

平成19年度 博士論文

FPGAを用いた聴覚情報処理システムの
ハードウェア実装に関する研究

名古屋工業大学 情報工学専攻
平成17年度入学 15517502番

岩佐 要

目次

第1章	はじめに	1
1.1	音による周辺環境把握とその実現	1
1.2	パルスニューロンモデルによる聴覚情報処理システム	2
1.3	本論文の構成	3
第2章	脳及び聴覚における情報処理	5
2.1	ニューロン	5
2.1.1	神経の興奮とその伝導	5
2.1.2	シナプス	7
2.2	聴覚器官	10
2.3	上オリーブ核複合体	14
第3章	聴覚情報処理システム	18
3.1	パルスニューロンモデル	18
3.2	周波数-パルス変換部	19
3.2.1	蝸牛モデル	20
3.2.2	有毛細胞モデル	23
3.2.3	パルス変換部	26
3.3	時間差検出部	26
3.4	音圧差検出部	28
3.5	ベクトル量子化部	30
3.6	パターン識別部	32
第4章	聴覚情報処理システムのハードウェア実装	33
4.1	FPGA(Field Programmable Gate Array)	33
4.2	パルスニューロンモデルの実装	35
4.3	時間差検出部の実装	36
4.4	音圧差検出部の実装	36

4.5	競合学習ネットワークの実装	38
第5章	音源定位及び音源種類識別実験	42
5.1	システム構成	42
5.2	実験条件	43
5.3	時間差検出結果	48
5.4	音源定位実験	50
5.5	音源種類識別実験	52
5.6	システムの耐雑音性	52
5.7	処理時間及び回路数の比較	54
5.8	まとめ	56
第6章	複数音源分離及び音源種類識別実験	58
6.1	システム構成	58
6.2	実験条件	59
6.3	時間差検出による音源分離	59
6.4	分離した各音源の音源種類識別実験	63
6.5	まとめ	66
第7章	音源接近検出及び音源種類識別実験	68
7.1	システム構成	68
7.2	実験条件	69
7.3	音圧差検出実験	72
7.4	音源種類及び接近識別実験	75
7.5	実際の音に対する認識実験	75
7.6	処理時間及び回路数の比較	80
7.7	まとめ	82
第8章	あとがき	83
	謝辞	86
	参考文献	88

表 目 次

3.1	各チャンネル番号 (Ch. No.) に対する中心周波数の値 (サンプリング周波数=48kHz)	23
3.2	各チャンネル番号 (Ch. No.) に対する中心周波数の値 (サンプリング周波数=16kHz)	24
5.1	使用する機器の仕様	45
5.2	音源定位及び音源種類識別実験におけるパラメータ	46
5.3	FPGA 上での各回路の回路数	46
5.4	各入力音に対する音源位置識別部の出力結果	50
5.5	音源種類識別部の出力結果	52
5.6	各部の処理時間の比較	54
5.7	ハードウェア実装時に並列化が可能な演算個数	54
5.8	各手法における並列化した際に用いる演算回路の個数	56
6.1	音圧差検出実験における各パラメータ	61
6.2	複数音源分離実験における CONP のパラメータ	63
6.3	音源種類の識別結果	64
7.1	音源接近検出及び音源種類識別実験におけるパラメータ	71
7.2	FPGA 上での各回路の回路数	71
7.3	音源種類と接近識別部の認識結果	75
7.4	各部の処理時間の比較	80
7.5	各手法における並列化した際に用いる演算回路の個数	81

目 次

2.1	ニューロンの構造 (脳・神経の科学 I[24] より)	6
2.2	膜電位の変化と活動電位	7
2.3	シナプスの構造	8
2.4	聴覚抹消系 (An Introduction to the Physiology of Hearing[1] より)	11
2.5	蝸牛の断面図 (An Introduction to the Physiology of Hearing[1] より)	13
2.6	有毛細胞の入出力関数 (An Introduction to the Physiology of Hearing[1] より)	14
2.7	Jeffress のモデル	15
2.8	位相多義性 ([29](平田ら) より)	16
3.1	パルスニューロンモデル	19
3.2	蝸牛モデルにおけるフィルタの構成	20
3.3	バタワース型フィルタの周波数-振幅特性	24
3.4	3次バタワース型フィルタ	25
3.5	時間差検出ネットワーク	27
3.6	時間差検出ネットワークにおけるパルス削除アルゴリズム	28
3.7	音圧差検出ネットワーク	29
3.8	パルスニューロンモデルのための競合学習ネットワーク	30
4.1	FPGA の基本構造	34
4.2	FPGA における電位の減衰の表現	36
4.3	FPGA におけるパルスニューロンモデル	37
4.4	FPGA における時間差検出ネットワーク	37
4.5	FPGA における LSO モデル	38
4.6	FPGA における競合学習ネットワーク	39
4.7	FPGA における競合学習ニューロン	40

4.8	FPGA における制御ニューロン	40
4.9	FPGA における入力パルス頻度を計算するニューロン	41
4.10	FPGA における閾値制御を行うユニット	41
5.1	音源定位及び音源種類識別を行うシステムの概略	43
5.2	音源定位実験における音源の移動	44
5.3	音源種類識別実験のための学習音	45
5.4	本研究で使用した FPGA ボード	47
5.5	時間差検出部の出力	49
5.6	音源位置識別部における出力結果	51
5.7	音源定位部の耐雑音性	53
5.8	音源種類識別部の耐雑音性	53
6.1	複数音源分離及び音源種類識別を行うシステムの概略	59
6.2	提案システムの動作例	60
6.3	実験における音源位置の状況	60
6.4	複数の音を与えた場合の時間差検出部の出力結果	61
6.5	各時間差における出力の周波数パターンの抽出	62
6.6	各時間差における識別部の出力頻度	64
6.7	学習後の各ニューロンの重み値	65
6.8	二方向に同じ音を用いた場合の時間差検出部の出力結果	66
7.1	音源接近検出及び音源種類識別を行うシステムの概略	69
7.2	学習実験に用いた音源の移動	70
7.3	各時間毎における音圧差検出部の出力結果	73
7.4	各入力音毎における音圧差検出部の出力結果	74
7.5	録音マイクの取付位置	76
7.6	路上音の録音状況	76
7.7	路上音における出力結果	79

第1章

はじめに

1.1 音による周辺環境把握とその実現

我々人間が普段生活する中で周辺の環境を把握する際には視覚や聴覚からの情報が大きな役割を持つ。特に聴覚から得られる情報量は大きく [1]，障害物の多い場所や夜間などにおいては視覚では得られない情報を得ることができる。さらに左右の両耳のみが利用されているという点も重要であり，二つの集音器のみで音源がどこにあるか (音源定位)，音源がどのような音であるか (音源種類の識別) を聞き分けることができることを示している。例えば左右水平方向の音源定位に関しては，左右の耳の間に生じる音の時間差や音圧差を用いているとされている [2]。このような人間の聴覚からの音情報処理能力を工学的に応用することができれば，聴覚障害者への支援や危険を通知する装置など様々な利用が期待できる。

障害者支援や危険通知のための装置を実用化する上で留意すべき点としては，装置の規模や入力信号に対する応答速度などがあげられる。

音源定位については精度よく定位を行う方法としてマイクロホンアレイを利用した研究が行われている [3][4][5][6]。しかしマイクロホンアレイを用いた場合集音器の数が多く大きな装置になるため，日常生活で使用するには不便である。他にマイクをロボットに搭載し，カメラからの画像とあわせて精度良く識別する研究も行われている [7][8]。また音源種類の識別に関しては，従来より数多くの研究がなされ集音器一つと計算機を利用した方法が一般的に用いられている [9][10]。しかし音源定位や画像処理などの他の処理を同時に行う場合には，高性能な計算機が必要となる。

音源の方向や種類を“識別する”という点については，パターン認識の分野では人間の脳細胞をモデル化したニューロンモデルが広く用いられている。しかし音のような時系列情報を扱うにはパターンの時間的な変化や順

序を許容することが必要であるため、各時間毎の入力データに対して静的に取り扱う従来のニューロンモデルは音情報処理には不向きである。時間情報を空間的に展開して扱う TDNN[11] などの様々な形のニューラルネットワークが使われているが、構成の複雑さに伴って使用する素子も多くなるため計算量も膨大になり、入力信号に対する十分な応答速度が得られない可能性がある。またニューロンモデルの他に識別を行う手法としては、音源定位では CSP 法 (Crosspower Spectrum Phase analysis)[5][12], 音源識別では GMM(Gaussian Mixture Model) を用いた方法 [13][14] など認識精度のよい手法が個別に提案されている。しかしこれらは積算を用いた計算が多く、ハードウェア化する場合には DSP などのコストの高い回路が必要となる。また各識別手法を個別に実装する場合、それぞれに対して専用の回路を用意する必要があり設計・製作が容易ではなくなる。このように、障害者支援や危険通知のための装置を実用化するには、識別手法の複雑さやそれに伴う回路規模の問題、装置そのものの大きさの問題など多くの課題が残されている。

1.2 パルスニューロンモデルによる聴覚情報処理システム

従来のニューロンモデルに対して、音のような時系列情報を扱うために、入出力信号をパルス列に限定したパルスニューロン (Pulsed Neuron 以下 PN) モデルが提案されている [15][16]。PN モデルでは時間的に減衰する膜電位で情報を表現するため時系列情報を扱うのに適しており、入出力がパルス情報であり演算が単純なため簡単な電子回路によって実装が可能であるという利点を持つ [17][18]。また、基本素子である PN モデルのハードウェア化が容易なため、PN モデルの組み合わせで実現されるシステム全体を一つのハードウェアに搭載することが可能となり従来手法より小規模なシステムを構築することができる。さらに各 PN モデルはハードウェア化することで並列に計算することができるため、高性能な計算機などを用いなくても高速な処理能力が期待できる。このような期待から、数多くの実装のためのパルスニューロンモデルが提案されている [19][20]。これらのモデルは確率パルス幅変調に基づく1ビット算術演算で構成されているが、確率パルス幅変調は量子化ノイズが大きく、統計的な性質が理想でない場合に演算誤差が大きくなるという問題がある。これに対して $\Delta\Sigma$ 変調を用いた量子

化誤差の少ない信号処理によるパルスニューロンモデルの実装が提案されている [21]. このモデルでは各ニューロン間の信号を1ビット幅で行うことができる半面, 高速な回路を必要とすることになる. 本研究では単純で安価な回路により聴覚情報処理システムの実現を試みるため, 素子の規模が大きくなるが各ニューロン間の信号は複数のビット幅を用いることにする.

なお本研究ではハードウェアへの実装にはField Programmable Gate Array(FPGA)を用いる. FPGAはプログラムにより利用者が独自の論理回路を自由に書き替えることのできるLSI(Large Scale Integration:大規模集積回路)の一種である. 近年になりFPGAの高速化, 大規模化が実現され, さらにCPUやSRAMなどを搭載し複雑なシステムを一つの開発ボードで実現できるようにしたものもある. FPGAの普及により計算機を用いて内部設計を行うだけで複雑なデジタル回路を容易に実現できるようになってきた [22].

また音情報を用いて周辺環境を把握する方法として, 黒柳らによってPNモデルを用いた聴覚情報処理モデルが提案されている [23]. このモデルでは人間の聴覚情報処理機構を模倣し, 二つの集音器から得られる音信号を複数の周波数帯域に分解した後, 各振幅値に比例した値でパルス列に変換する. そして各周波数毎のパルス列から周波数特性や時間差などの特徴量を用い音源定位や音源種類の識別などを行う. このうちパルス列に変換した後の特徴量の抽出や識別などは全てPNモデルで行われる. 各部を同じPNモデルを基本素子とし, 異なる機能を持つ回路をPNモデルを組み合わせることで構成する. これにより識別手法毎に個別の回路を製作する必要がなく一つの装置にまとめることが容易であるため, 最終的な装置の規模を小さくできる. 加えて基本素子が同じであるため併用時には各識別結果を相互に反映させることが容易であり, 性能の向上も期待できる.

以上のことから, 本研究ではFPGA上に多数のPNモデルを実装し, これらを組み合わせることで音源定位や音源種類の同定などの聴覚情報処理を同時に行うことのできる聴覚情報処理システムを提案する.

1.3 本論文の構成

本論文ではまず2章にて提案モデルの基となる蝸牛や有毛細胞などの人間の聴覚における各部位の構成と機能について述べる. また生体の神経細胞

において情報となるパルスの発生とその影響について述べる。

続いて3章にて人間の聴覚を基にした聴覚情報処理システムについて説明する。

4章では3章で説明した聴覚情報処理システムのうち、PNモデルによって構成される部分をFPGAを用いてハードウェア上に実装する方法について述べる。

5章では、ハードウェア実装したPNモデルを組み合わせ、音源定位と音源種類の同定を行う聴覚情報処理システムを構成する。また実験より、構成した聴覚情報処理システムが音源定位と音源種類の識別を同時に行うことができ、従来手法と比較して少ない回路数で実現できることを示す。

6章では、実際の環境における識別のためにPNモデルを用いて複数の音源を分離し個々の音源に対して識別を行う方法を説明する。

7章では、同じくハードウェア実装したPNモデルを組み合わせ、音源の接近検出と音源種類の識別を行う聴覚情報処理システムを構成する。そして構成した聴覚情報処理システムが音源の接近検出と種類の識別を同時に行うことができることを示す。

以上のように本論文ではFPGAを用いてハードウェア実装した聴覚情報処理システムを提案し、各種実験より提案システムが様々な聴覚情報処理を高速に行うことができ、従来手法より少ない回路規模で実現できることを示す。本研究で提案した聴覚情報処理システムにより、障害者支援や危険通知のための装置を小規模で実用化することが期待できる。

第2章

脳及び聴覚における情報処理

2.1 ニューロン

人間の脳には千億以上の神経細胞が存在している。ニューロン (neuron) とはこの神経細胞のことでドイツの Waldeyer によってニューロンという用語が導入された [24]。隣接したニューロン間には連続した接続は存在しないが、互いに交錯することで複雑な回路を形成し、信号を送ることによって情報処理を行なっている。

図 2.1 は 1 個のニューロンの構造を示す。全てのニューロンには細胞体 (cell body) と神経繊維 (軸索, axon) が存在し、細胞体からは軸索の他に多数の枝分れを持つ数本の突起が派生している。これらの突起はその形状から樹状突起 (dendrite) と呼ばれる。一方、軸索は細胞体から数百 μm 離れた箇所からは髄鞘 (ミエリン鞘, myelin sheath) で覆われ、終端部分が他の素子の樹状突起と結合しており、この部位をシナプス (synapse) と呼ぶ。

2.1.1 神経の興奮とその伝導

膜電位

他の全ての細胞と同様に、ニューロンの細胞体とその軸索には細胞膜 (cytoplasmic membrane) が存在する。細胞膜の内外での電位差のことを膜電位 (membrane potential) と呼ぶ。

細胞膜はリン脂質からなる半透膜であり、 K^+ のような小さなイオンと水分子に透過性を示す。細胞内液と細胞外液の浸透圧は同一であるが、細胞膜内外ではイオンの組成が異なり、細胞内では K^+ 濃度が高く、細胞外では Na^+ および Cl^- 濃度が高い。その濃度勾配により膜透過性を持つ K^+ は細胞外へ流出する傾向があり、流出すると細胞内は正の電荷を失うため細胞内は細胞外に対して負の電位となる。負の電位となると逆に K^+ を細胞内へ

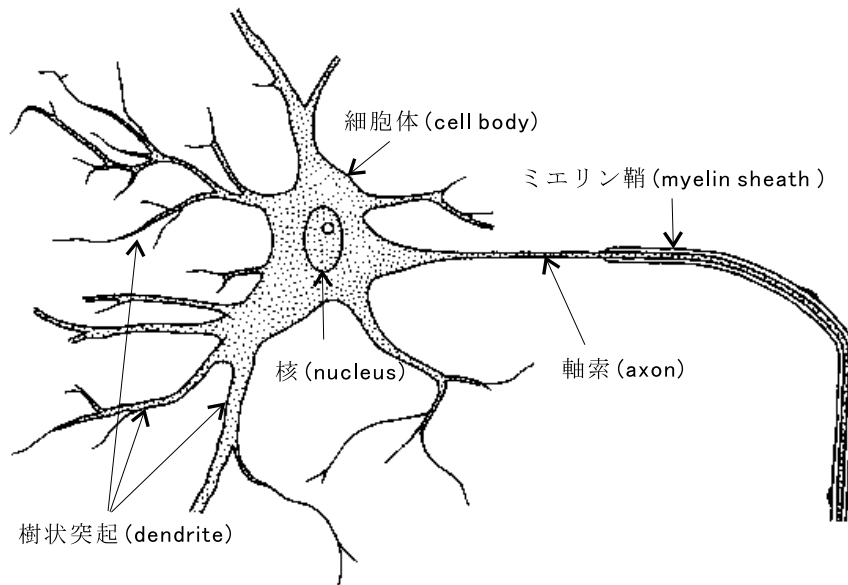


図 2.1: ニューロンの構造 (脳・神経の科学 I[24] より)

引き戻す力が働くため，結果として細胞膜においては K^+ の濃度勾配による外向きの力と K^+ の流出による内向きの電位勾配による力が均衡を保つ状態になる．この均衡状態では，細胞膜内の電位は細胞膜外に対して約 $-90mV$ でありこの膜電位をニューロンの興奮していない時の膜電位という意味で静止膜電位 (resting membrane potential) と呼ぶ．

電気刺激により細胞内がプラス，細胞外がマイナスになるように外向きの電流を流すと静止膜電位が減少してニューロンの興奮を誘発する．膜電位は細胞膜内外の電气的分極のことであるため，静止膜電位の減少のことを脱分極 (depolarization) と呼び，逆に細胞内外で内向きの電流を流すと膜電位がさらに負の方向に増加し，この増加のことを過分極 (hyperpolarization) と呼ぶ．

活動電位

ニューロンがある電位 (例えば $-90mV \sim -30mV$) まで脱分極すると細胞膜の Na^+ に対する透過性が高まり，細胞内に流入するようになる．細胞内の Na^+ 濃度が増加するとさらに脱分極が進み，最終的には細胞内の電位は細胞外に対して正の電位を持つ．この膜電位を活動電位と呼び，約 $+50mV$ である．また，このような時間に対する電位の変化は図 2.2 に示され，急激な電位の上昇のことをニューロンの発火 (firing)， Na^+ が流入するようにな

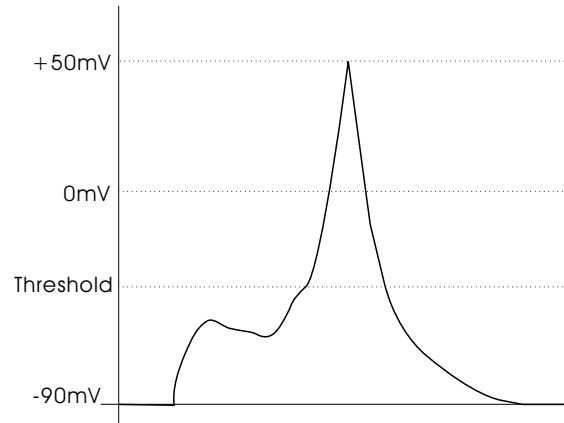


図 2.2: 膜電位の変化と活動電位

る電位を閾値 (threshold) と呼ぶ.

しかし細胞の脱分極が持続することによる Na^+ の透過性の低下と細胞の脱分極により Na^+ に少し遅れて K^+ に対する透過性の増加との2つの要因から活動電位は約 1msec しか持続しない. K^+ の透過性が増加すると K^+ が急激に流出し, 細胞内の電位が再び静止膜電位に回復する. 一度興奮して K^+ が流出してから, 再びある程度の量に戻るまでにはある程度の時間を必要とするため, 次の活動電位の発生には時間がかかる. この時間を不応期 (refractory period) と呼ぶ.

イオンチャネル

Na^+ と K^+ はイオン組成が異なり, さらに細胞内外はリン脂質の二重層で形成された細胞膜によって隔てられているため, 細胞内外を自由に透過できないことは明らかである. しかし細胞膜にはイオンチャネル (ion channel) と呼ばれるイオンの通過経路があり, Na^+ と K^+ のそれぞれが専用のイオンチャネルである Na^+ チャネルと K^+ チャネルを独立で通過することができる. 静止膜電位の状態では Na^+ チャネルと K^+ チャネルは開く確率が低い, 脱分極により開く確率が増大し閾値に達すると自由に通過できるようになる.

2.1.2 シナプス

シナプスの構造

シナプスの構造を図 2.3 に示す [25].

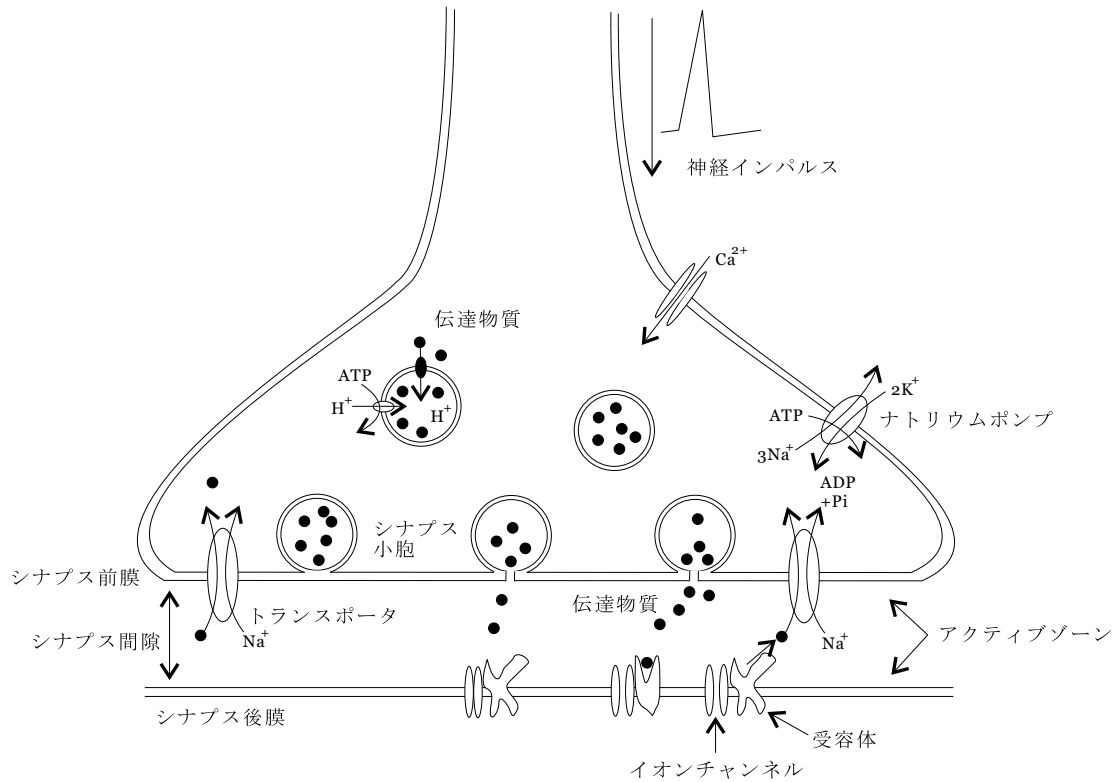


図 2.3: シナプスの構造

シナプス前細胞の軸索の終末端とシナプス後細胞の間の間隔をシナプス間隙 (synaptic cleft) と呼ぶ。シナプス間隙は約 $10 \sim 100nm$ である。シナプス前細胞の末端部の細胞膜をシナプス前膜 (presynaptic membrane), シナプス後細胞のシナプス部の細胞膜をシナプス後膜 (postsynaptic membrane) と呼び、ここで伝達物質 (neurotransmitter) のやりとりをする。また、シナプス前細胞の末端部にはシナプス小胞 (synaptic vesicle) という小さな穴が空いており、そこに伝達物質が含まれている。

シナプスでの情報伝達

シナプスにおける情報伝達は以下のステップで行なわれる。

1. 神経パルスの伝播 細胞体の電位が閾値を超えると活動電位 (神経インパルス) が発生する。活動電位は軸索上を伝播し、シナプス終末端に到達する。
2. シナプス終末端でのイオンチャンネル開閉 活動電位によりシナプス終末端が脱分極する。その結果、シナプス終末端の電位依存性 Ca^{2+} チャンネルが開き、 Ca^{2+} の流入が起こる。この時、 Ca^{2+} チャンネル近傍では Ca^{2+}

が局所的に高濃度になると考えられる。

3. シナプス小胞, 伝達物質の放出 Ca^{2+} が流入すると, 神経伝達物質を含むシナプス小胞がシナプス終末端に向かって動き出し, 小胞の中に入っていた伝達物質が間隙に放出される. このプロセスはエキソサイトシスと呼ばれる.
4. 伝達物質の拡散 放出された伝達物質が拡散し, シナプス後膜に達する.
5. 伝達物質の受容体への結合 シナプス後膜に達した伝達物質は, 膜上の受容体 (receptor) と結合する.
6. イオンチャネルの開閉 受容体に伝達物質が結合すると, イオンチャネルの開閉が制御される.
7. 伝達物質の回収, シナプス小胞の再利用 シナプス前膜などには, 放出された伝達物質を速やかに細胞内に取り込み, 神経伝達を終了させるトランスポータが存在している. また, シナプス間隙には伝達物質を分解する酵素も存在する. 回収された伝達物質は, 再び伝達物質として再利用される. シナプス終末端には伝達物質をシナプス小胞内に輸送, 濃縮するメカニズムも存在する.

シナプス後電位

チャネル制御のメカニズムは大きく2種類に分けられ, 伝達物質受容後に直接チャネルの開閉が制御されるイオンチャネル直結型と, 細胞内に細胞内伝達物質を放出し, チャネル開閉を制御する代謝調節型がある. チャネル開閉の結果, シナプス後細胞にはシナプス後電位が発生する. シナプス後膜で脱分極を起こし発火に近づける興奮性シナプス (excitatory synapse) で発生するシナプス後電位をそれぞれ興奮性シナプス後電位 (excitatory postsynaptic potential:EPSP), 過分極を起こし発火から遠ざける抑制性シナプス (inhibitory synapse) で発生するシナプス後電位を抑制性シナプス後電位 (inhibitory postsynaptic potential:IPSP) と呼ぶ.

なお伝達物質の効果は, 伝達物質により制御されるイオンチャネルの種類や膜電位の状態などに依存するものであり, 同じ伝達物質が興奮性として働く場合もあれば, 抑制性として働く場合もある.

シナプスの加重

実際にはシナプス前細胞からの活動電位によるシナプス後電位の変化はあまり大きくなく、発火をするためには複数のシナプス結合による脱分極での和が必要となる。他に、一回の活動電位によるシナプス後電位の変化が回復する前に次の活動電位が到達すると、電位が加算されて閾値を超える場合もある。これらのように異なる活動電位による膜電位の加算により大きな膜電位となることを加重といい、前者のように複数の異なるシナプスによる加重を空間加重、後者のように時間的に活動電位が連続することによる加重を時間加重という。

シナプス修飾作用

神経回路に情報が伝達するかどうかは、膜電位が閾値に達して活動電位が発生するかどうかによって決定される。シナプス修飾作用 (synaptic modulation) とはこのシナプスにおける伝達効率を調節する作用のことである。シナプス修飾によってシナプス電位の大きさが変化すると、神経回路が開くか閉じるかが方向付けられる。例えば、元々は EPSP が活動電位の閾値に達しないために、以降の回路へ信号が伝わらないようなシナプスが、修飾を受けて EPSP が増大して閾値に達したような場合、信号は活動電位を介して次のニューロンに伝達され、従来閉鎖していた神経回路が活動することがある。逆にシナプス修飾により神経回路が閉じることもあり、その影響は無視できない。シナプス修飾の持続時間は msec 程度の短いものから何年にも及ぶ長いものまである。特に長時間持続するシナプス修飾は、シナプス可塑性 (synaptic plasticity) と呼ばれ、学習記憶の基本メカニズムと考えられている。

2.2 聴覚器官

図 2.4 に聴覚器官の概観を示す。聴覚器官は外耳 (External ear), 中耳 (Middle ear), 内耳 (Internal ear) から成り立っており、空気の振動という物理的信号が電气的信号に変換される [26][27]。

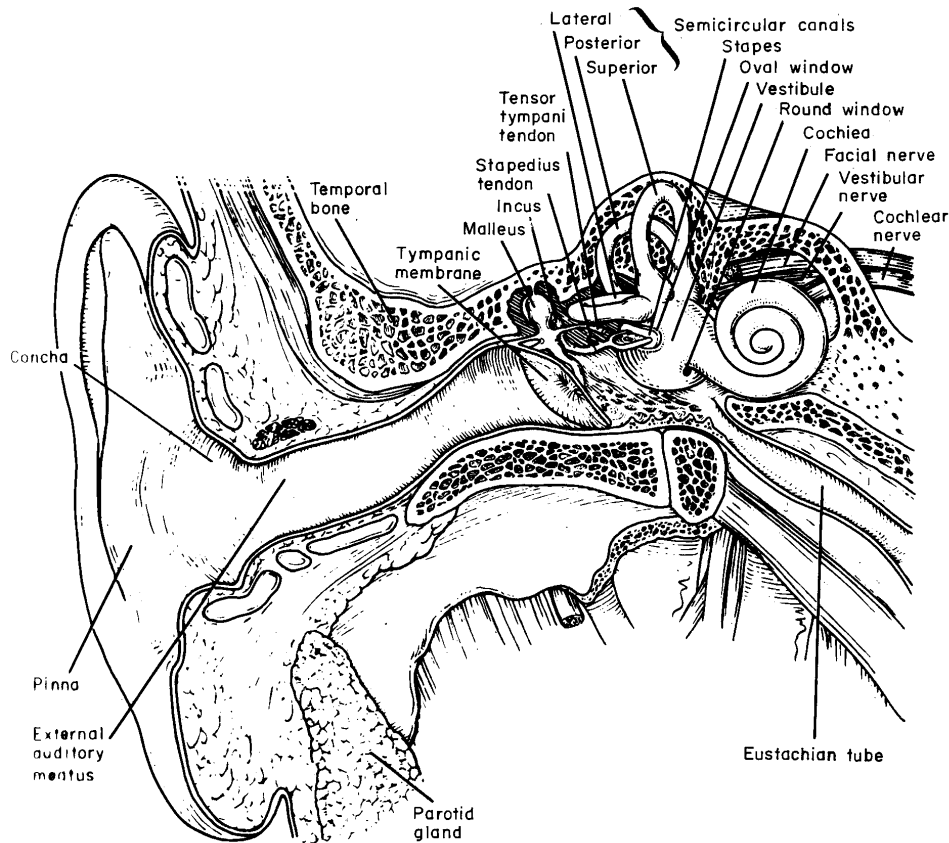


図 2.4: 聴覚抹消系 (An Introduction to the Physiology of Hearing[1] より)

外耳

音は漏斗状の耳介 (pinna) によって収集され、耳甲介 (concha)、外耳道 (external auditory meatus) を通過後、鼓膜を振動させる。この耳介から鼓膜までの部分を外耳と呼ぶ。耳甲介は約 5kHz の周波数成分を共鳴によって増幅する。外耳道の長さは約 25mm で 3430Hz の 1/4 波長に相当しているので、3 ~ 4kHz の周波数成分を増幅する (音速を 343m/sec として計算)。結果的に外耳によって 2kHz から 7kHz の周波数が増幅され、その圧力は外耳道の外の圧力の約 2 倍となる。

また人間の頭の直径は約 18cm であるので、18cm よりも長い波長である 2kHz よりも低い周波数の音は反対側の外耳にも到達するが、それよりも高い周波数の音は頭によって反射してしまい反対側の外耳には到達しにくい。

中耳

外耳から鼓膜に伝わった振動は、まず鼓膜と接着しているつち骨 (malleus) に伝わり、次にきぬた骨 (incus)、あぶみ骨 (stapes) を介して蝸牛管へ伝わる。この部分を中耳と呼び、つち骨、きぬた骨、あぶみ骨をまとめて耳小骨 (auditory ossicles) と呼ぶ。この耳小骨の連鎖はこの作用によって外耳の音圧を約 22 倍 (27dB) 増幅して空気と蝸牛管内のリンパ液とのインピーダンスの不一致を軽減する。つち骨には鼓膜張筋 (三神経支配)、あぶみ骨にはあぶみ骨筋 (顔面神経支配) とよばれる筋がついており、強大音を聴くと、反射によりこれらの筋が収縮して中耳伝音系のインピーダンスを高め、音を内耳に伝えにくくする。

内耳

内耳は側頭骨錐体部にある骨でつくられた器官で、前方から蝸牛 (cochlea)、前庭 (vestibule) 及び三半規管 (semi-circular canals) からなっており、互いに交流する Na^+ 濃度の高い外リンパ液を満たしている。内耳の開口部は前庭窓 (oval window) と蝸牛窓 (round window) で、前庭窓はあぶみ骨底によって蓋をされ、蝸牛窓は第二鼓膜と呼ばれる薄い膜によって閉じられている。実際に聴覚と関係があるのは蝸牛であり、かたつむりの殻に似た形をし円錐形の蝸牛軸を中心にして約 $2\frac{3}{4}$ 回転している骨性の管である。その螺旋状の管の長さは約 35mm であるが、内部は蝸牛軸から突出した薄い骨の板と基底膜 (basilar membrane) によって二分されている。蝸牛の断面を示した図が図 2.5 である。

下部の管は鼓室階 (scala tympani) と呼ばれ、一端は蝸牛窓で中耳に接し、上部の管は前庭階 (scala vestibuli) で一端は前庭窓であぶみ骨と接している。鼓室階と前庭階の他端は螺旋管の先端、蝸牛孔 (helicotrema) でつながっており、中の外リンパ液は両階を自由に流通している。前庭階の一部は薄いライスネル膜 (Reissner's membrane) によって仕切られて蝸牛管を形作っている。蝸牛管は内リンパ液で満たされており、その中に複雑な働きをする聴覚受容器のコルチ器 (organ of corti) がある。

基底膜は先端にいくほど幅が広がっており、逆に蝸牛自体は先端にいくほど直径が小さくなっている。このため基底膜は、高い音の場合は基部の側がよく振動し、低い音の場合は先端部がよく振動するので、蝸牛は音の振

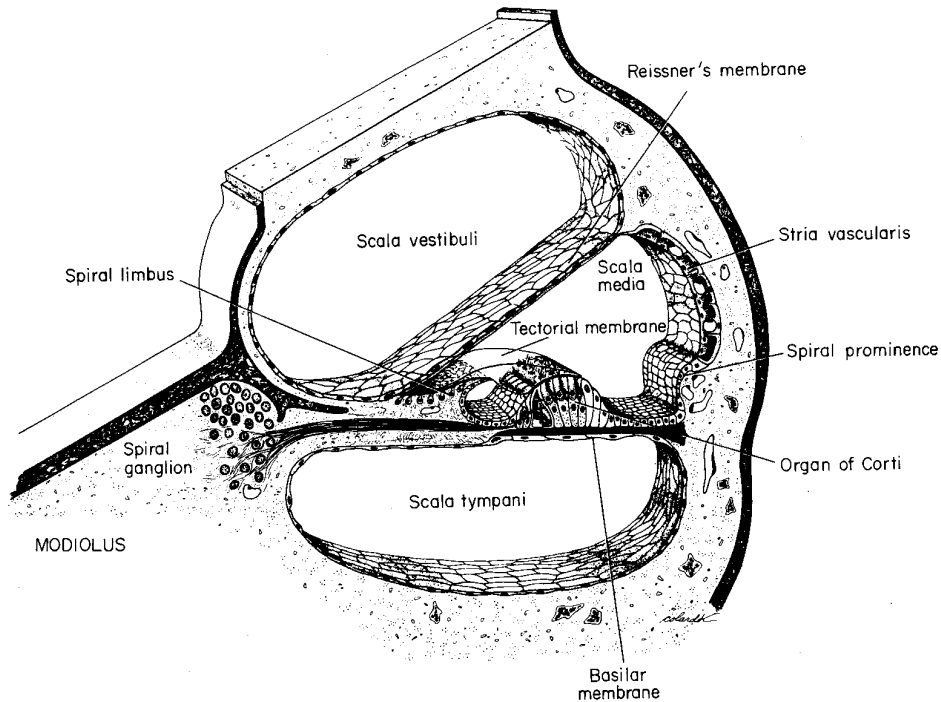


図 2.5: 蝸牛の断面図 (An Introduction to the Physiology of Hearing[1] より)

動を周波数成分ごとに分解する役割をもつと考えることができる。

有毛細胞

蝸牛管と鼓室階を分けている基底膜の上には、三角状になった柱細胞 (rods of corti) の両側に 1 列の内有毛細胞と 3 ~ 4 列の外有毛細胞が並んでいる。基底膜が振動すると網状板 (reticular lamina) はコルチ柱に支持され、基底膜は蝸牛軸骨との接続部を軸にして動くが蓋膜 (tectorial membrane) は螺旋縁 (limbus) との接続部を軸として動くので、その間に機械的なずれが生じる。内有毛細胞の先端の毛の一部は、網状板を通り蓋膜に食い込んでいるので機械的なずれによって毛が曲げられることになり内有毛細胞が刺激され、電気生理的な電位の形で聴覚神経の興奮をもたらすと考えられている。外耳中耳を通過してきた機械的振動がここで電氣的信号に変換され、聴覚神経系への入力信号が生じることになる。この内有毛細胞の刺激による興奮によって生じる膜電位変化は短い毛から長い毛の方向に刺激があった時のみ起こる。正弦波振動による刺激の大きさに対する膜電位の変化は図 2.6 のようになっており整流特性を持っている。また外有毛細胞は基底膜の振動に対して能動的に振動し、周波数特性を鋭くしているといわれている。

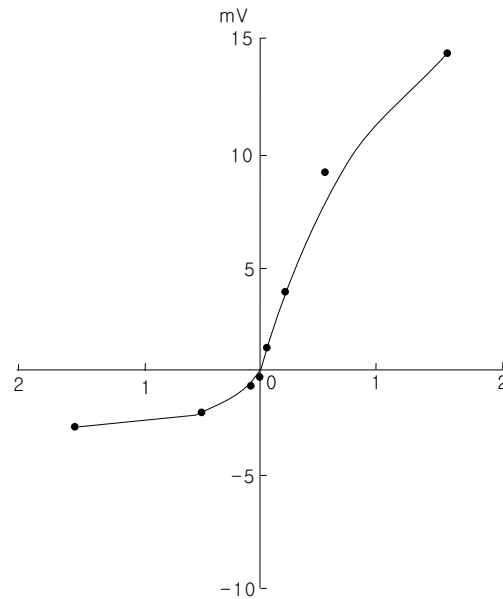


図 2.6: 有毛細胞の入出力関数 (An Introduction to the Physiology of Hearing[1] より)

蝸牛神経

有毛細胞によって電気的な信号に変換された音情報は蝸牛神経 (cochlear nerve) に伝達されインパルス列へと変換される。蝸牛神経は細胞体から両方向に線維が伸びている両極細胞であって、入力側の線維は有毛細胞に接続し、出力側の線維は上位の蝸牛背腹側神経核の細胞体にシナプス結合している。このため、有毛細胞の膜電位の上昇によって2.1.2で述べた動作と同様に伝達物質が伝達されて蝸牛神経へイオンが流入することで蝸牛神経の膜電位が上昇する。有毛細胞の膜電位の上昇が大きいほど、蝸牛神経も発火しやすくなるためインパルスの頻度によって音の大きさが伝えることができるようになる。

2.3 上オリーブ核複合体

蝸牛神経で発生したパルス情報は蝸牛神経核 (cochlear nucleus) を経て上オリーブ核複合体 (superior olivary complex : SOC) に伝達される。この段階で初めて左右の耳から入った信号が交差し、両耳信号処理、特に音源定位における重要な役割を果たしている。SOCは内側上オリーブ核 (medial superior olive : MSO)、外側上オリーブ核 (lateral superior olive : LSO)、台形体内側核 (medial nucleus of trapezoid body : MNTB) などから構成されており、MSOで両耳間の時間差 (Interaural Time Difference : ITD)、LSO

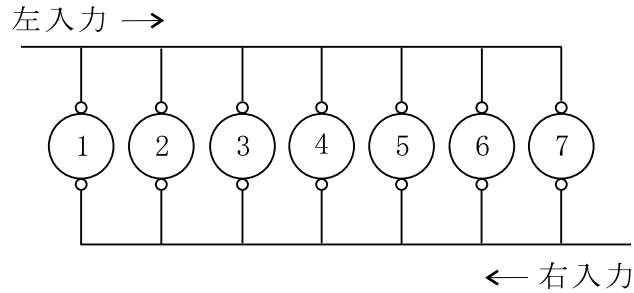


図 2.7: Jeffress のモデル

で音圧差 (Interaural Level Difference : ILD) を検出し, この情報を用いて左右方向の音源定位を行なっていると考えられている.

内側上オリーブ核

内側上オリーブ核 (MSO) 内のニューロンはシナプスが左右両方向に対称に延び, その多くは低周波音に反応し同側の前方腹側蝸牛神経核 (anteroventral cochlear nucleus : AVCN) と, 反対側の AVCN からの興奮性の入力を受けとる EE(excitatory-excitatory) ニューロンである. EE ニューロンは一方の刺激で興奮し, もう一方からの刺激で抑制を受ける EI(excitatory-inhibitory) ニューロンとは異なり, いずれの耳からの刺激に対しても興奮するニューロンである. MSO 内では左右からの入力は神経繊維の伝達速度やニューロンの空間的位置によって時間的な遅れをもって各ニューロンに入力される. 入力から離れたニューロンほど遅れて入力が到達するため, 左右からの入力の時間差により特徴的に反応することになる. MSO の振舞を説明するのモデルとして, 図 2.7 に示す Jeffress のモデルがある [28]. このニューロンは左右からの音が同時に入力されたときに発火する.

例えば左右の入力から同時にインパルス信号があった場合, 二つの信号は中央部分で同時に到達することになり, 図 2.7 中の 4 のニューロンが発火する. 左からのインパルス信号が先に入力された場合には左からの信号が多く移動することになるため同時に入力される箇所は中央より右にずれることになり, 5~7 のニューロンが時間差に応じて発火する. 逆に言えば, 発火したニューロンから左右の信号の時間差がわかることになる.

実際の聴覚系においては単一のインパルス信号ではなく各周波数に対応したインパルス列であり, その反応は異なる. 蝸牛神経は周期的な入力に対して phase-lock の性質を示し, 特定の位相で発火する確率が高い. このため

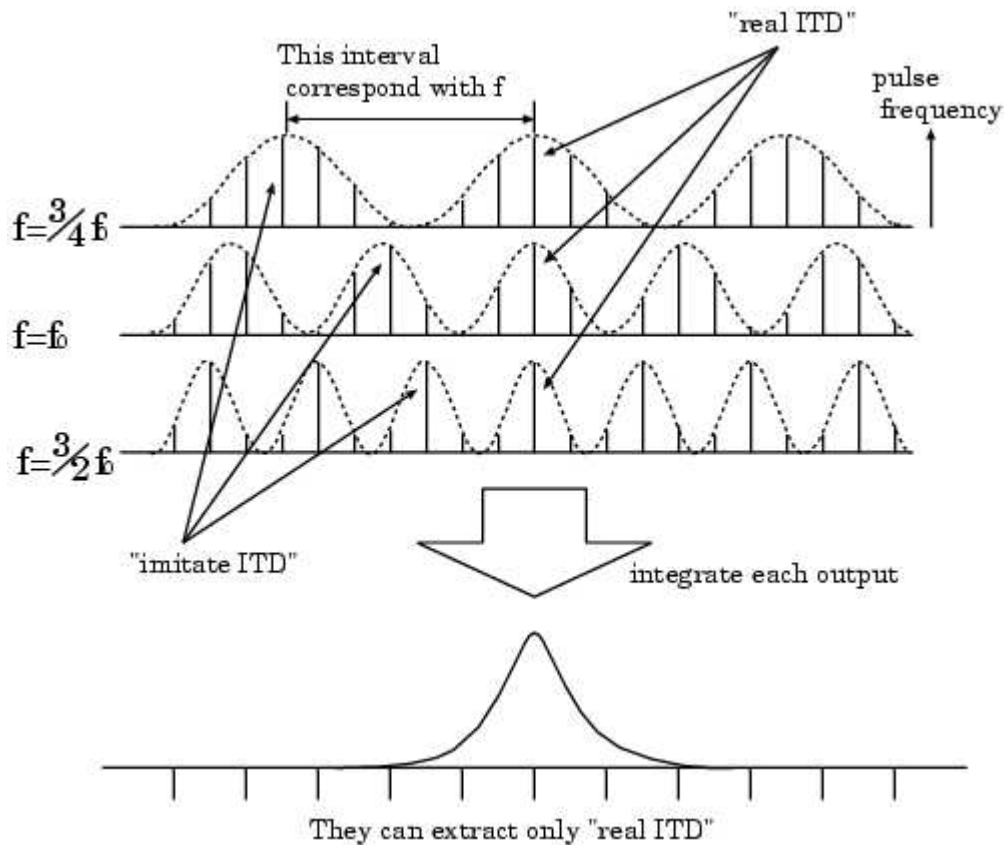


図 2.8: 位相多義性 ([29](平田ら) より)

入力されるインパルス信号に対しても特徴周波数の周期毎にインパルスの存在確率が高くなり、特徴時間差 T_{ITD} だけでなく、 $T_{ITD} \pm nT$ (T : 特徴周波数の周期) で現れる“偽の ITD”にも反応する (図 2.8 上部). これを位相多義性という.

しかし各特徴周波数によって周期が異なるため、偽の ITD の間隔が異なり実際の ITD に反応するニューロンの位置はどの周波数でも同じため、図 2.8 下部のように複数の周波数の情報を統合することによって“真の ITD”を検出することができる.

外側上オリブ核

外側上オリブ核 (LSO) には高周波数音によく反応するニューロンが多く、同側の AVCN から興奮性の入力を受け、反対側の AVCN からの入力を同側の台形体内側核 (MNTB) 経由でグリシン (glycine) 抑制性の入力として受ける EI ニューロンが大半を占めており、左右耳間の音の音圧差を検出する機能を持つと考えられている. LSO からの出力は両側の外側毛帯核

(nucleus of lateral lemniscus : NLL) と下丘 (inferior colliculus : IC) に送られている。音圧差を検出する機構については詳しいことはまだ解明されていないが、各ニューロンの抑制の程度は核内組織的に変化しており、音圧差の大きさに応じて LSO 内の発火する細胞の領域が変化することによって検出しているのではないかと考えられている。

第3章

聴覚情報処理システム

3.1 パルスニューロンモデル

音信号のような時系列データに関して情報処理を行う場合には、時間的な関係を扱うことができるモデルであることと、リアルタイム処理のために演算が簡単であることが望ましい。一般的に用いられている従来のニューロンモデルでは各単位時間における静的な入力ベクトルのみを扱うため時系列情報のような動的な入力を扱うにはフィードバックを用いるなどのネットワーク構成を変更しなければならない、各素子における演算コストも高くなる。これに対して、パルスニューロン (Pulsed Neuron:以下PN) モデルが提案されている [15][30]。PN モデルと通常のニューロンモデルとの違いは、PN モデルが入出力に神経パルスを想定し0か1の値をとるパルス信号を用いている点である。このため各素子における演算が単純であり、大規模なネットワークを少ないハードウェア資源で構成することができる。また入力の情報を時定数で減衰する膜電位として保持している点も通常のニューロンモデルと異なり、音信号のような時系列信号の扱いに適している。

PN モデルの構造を図3.1に示す。PN モデルは他のニューロンと結合する k 本のシナプス、各シナプスに対する結合重み w_k 、各シナプスにおける局所膜電位 $p_k(t)$ 、各局所膜電位の総和である内部電位 $I(t)$ 、出力を他のニューロンに伝達する軸索から構成される。時刻 t の値は連続値として扱うことも可能であるが、本研究では計算機上及びFPGA上でPNモデルを用いるために離散値とし、最小時間単位を $\Delta t[s]$ とする。サンプリング周波数を f_s とすると、 $\Delta t = 1/f_s$ である。PNモデルにおいては、まず各シナプス k において入力パルス $i_k(t)$ が入力されると、それぞれのシナプスに対応した結合重み w_k の値だけ局所膜電位 $p_k(t)$ が増加し、その後時間の経過とともに時定数 τ_k で指数関数的に減衰する (式 (3.2))。次にPNモデルは単位時間毎に

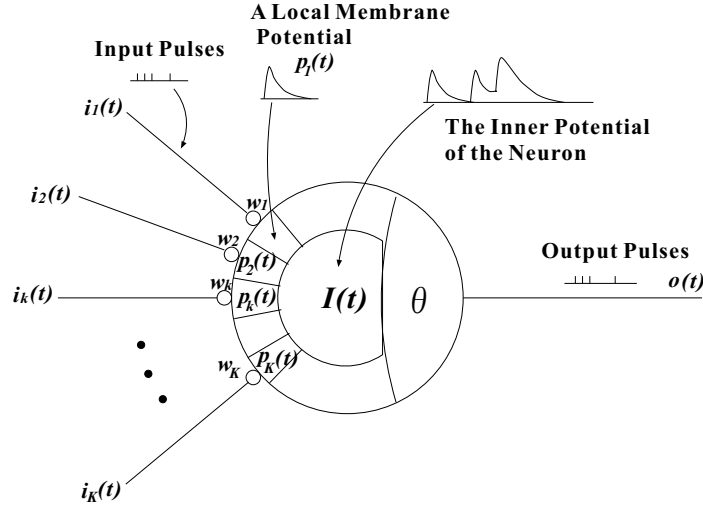


図 3.1: パルスニューロンモデル

局所膜電位 $p_k(t)$ の総和である内部電位 $I(t)$ を計算する (式 (3.3)). この内部電位 $I(t)$ が閾値 θ を越えると発火, 時刻 t において出力 $o(t)$ として 1 を出力する (式 (3.3)). またある PN モデルが一度発火した後, 一定時間 RP の間は内部電位が閾値を越えた場合はこの PN モデルは発火できない. この時間を不応期と呼ぶ.

$$p_k(t) = w_k i_k(t) + p(t - \Delta t) \exp\left(-\frac{\Delta t}{\tau}\right) \quad (3.1)$$

$$I(t) = \sum_{k=1}^K p_k(t) \quad (3.2)$$

$$o(t) = f_{usf}\left(I(t) - \theta\right) \quad (3.3)$$

3.2 周波数-パルス変換部

まず音信号は周波数-パルス変換部に入力される. 周波数-パルス変換部では, 左右の信号毎にまず周波数情報に変換された後, Band Pass Filter 群によって各中心周波数の特性を持った周波数帯域に分割される [31]. なお各フィルタはデジタルフィルタである [32]. 各フィルタの周波数特性は各フィルタの中心周波数が対数尺度において等間隔になるように決定してある. 次に各周波数帯の情報毎にパルス列に変換する. パルス列への変換は, 周波数変換後の振幅値が大きいほど確率的に多くのパルス列が含まれるようにする.

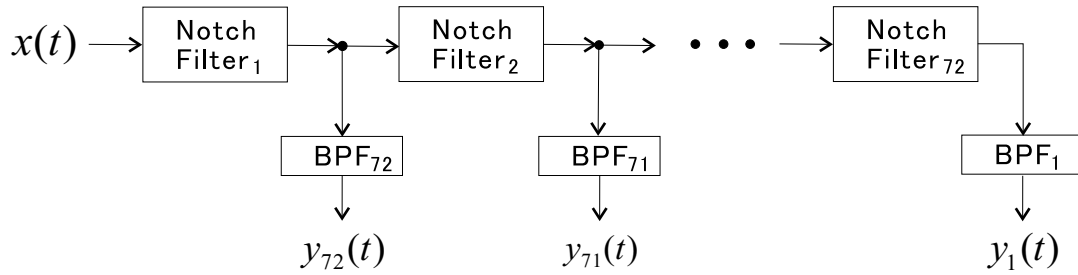


図 3.2: 蝸牛モデルにおけるフィルタの構成

3.2.1 蝸牛モデル

2.2 節で述べたように，蝸牛では入力された信号を各周波数成分毎に分解している．これは Band Pass Filter (BPF) 群を用いてモデル化することができる．本研究では蝸牛モデルでのフィルタとして Lyon の Cascade/Parallel-function 型フィルタモデルを用いる [31]．このモデルの構成は図 3.2 のようになっており，全部で 72 個縦続接続された 2 次 Notch Filter と各 Notch Filter の後に 2 次 BPF 並列接続されるという構成になっている．

また，本研究では高周波数領域での振幅補償を行うため，10kHz の周波数特性に FIR 型の Low Pass Filter (LPF) と 3 段の 2 次 Notch Filter を追加する．2 次 Notch Filter は低 $Q(6)$ の極と上側近傍の高 $Q(6 \sim 70)$ の零点からなる．BPF に関しては 0Hz の周波数に零点を持ち，前段と次段の Notch Filter の零点の間に高 Q の極をもっている．これによって高域側に急峻な遮断特性をもつ非対称 BPF となる．

Notch Filter の設計

s 平面における 2 次 Notch Filter の伝達関数は式 (3.4) により与えられる．

$$|T(j\omega)| = H \cdot \frac{s^2 + (\omega_z/Q_z) + \omega_z^2}{s^2 + (\omega_p/Q_p) + \omega_p^2} \quad (3.4)$$

ここで Q_z は零点での Q ， Q_p は極の Q ， ω_z は零点での共振角周波数， ω_p は極での共振角周波数である． H は低域での利得を 1 とするため式 (3.5) のようにする．

$$\begin{aligned} |T(j0)| &= H \cdot \left(\frac{\omega_z}{\omega_p}\right)^2 = 1 \\ H &= \left(\frac{\omega_p}{\omega_z}\right)^2 \end{aligned} \quad (3.5)$$

式 (3.4) を式 (3.6) で示される双一次 z 変換により変換すると、式 (3.7) になる。

$$s = \frac{2}{T} \cdot \frac{1 - Z^{-1}}{1 + Z^{-1}} \quad (3.6)$$

$$T(z) = \frac{a_2 \cdot z^{-2} + a_1 \cdot z^{-1} + a_0}{1 - b_2 \cdot z^{-2} - b_1 \cdot z^{-1}} \quad (3.7)$$

但し式 (3.8) の周波数変換が必要である。 ω_A はアナログフィルタの角周波数、 ω_D はデジタルフィルタの角周波数を示す。

$$\omega_D = \frac{2}{T} \tan\left(\frac{T}{2}\omega_A\right) \quad (3.8)$$

式 (3.7) より差分方程式は式 (3.9) とすることができる。

$$y(n) = a_2x(n-2) + a_1x(n-1) + a_0x(n) + b_2y(n-2) + b_1y(n-1) \quad (3.9)$$

式 (3.9) における定数は A を式 (3.10) と置くことにより式 (3.11) ~ 式 (3.15) によって求めることができる。ここで * は式 (3.8) により変換された後の値である。

$$A = 2^2 + \frac{\omega_p^*}{Q_p^*} \cdot 2T + (\omega_p^*T)^2 \quad (3.10)$$

$$a_0 = \frac{1}{A} \left(\frac{\omega_p^*}{\omega_z^*}\right)^2 \left\{2^2 + \frac{\omega_z^*}{Q_z^*} \cdot 2T + (\omega_z^*T)^2\right\} \quad (3.11)$$

$$a_1 = \frac{1}{A} \left(\frac{\omega_p^*}{\omega_z^*}\right)^2 \{-2^3 + 2(\omega_z^*T)^2\} \quad (3.12)$$

$$a_2 = \frac{1}{A} \left(\frac{\omega_p^*}{\omega_z^*}\right)^2 \left\{2^2 - \frac{\omega_z^*}{Q_z^*} \cdot 2T + (\omega_z^*T)^2\right\} \quad (3.13)$$

$$b_1 = -\frac{1}{A} \{-2^3 + 2(\omega_p^*T)^2\} \quad (3.14)$$

$$b_2 = -\frac{1}{A} \left\{2^2 - \frac{\omega_p^*}{Q_p^*} \cdot 2T + (\omega_p^*T)^2\right\} \quad (3.15)$$

また設計した Notch Filter においては $RI = F_p/F_z = 0.97$ で固定とし零点の Q は 30, 極の Q は 6 ~ 70 とする。

BPF の設計

s 平面における 2 次 BPF の伝達関数は式 (3.16) により与えられる。

$$|T(j\omega)| = H \cdot \frac{(\omega_b/Q_b)s}{s^2 + (\omega_b/Q_b)s + \omega_b^2} \quad (3.16)$$

ここで Q_b は BPF の Q , ω_b は零点での共振角周波数を表す。 H は低域での利得を 1 とするため、式 (3.17) のようにする。

$$|T(j\omega_b)| = H = 1 \quad (3.17)$$

式 (3.16) を式 (3.6) の双一次 z 変換により変換すると, 式 (3.18) になる.

$$T(z) = \frac{a_2 \cdot z^{-2} + a_1 \cdot z^{-1} + a_0}{1 - b_2 \cdot z^{-2} - b_1 \cdot z^{-1}} \quad (3.18)$$

但し式 (3.8) の周波数変換が必要である. 式 (3.18) より差分方程式は式 (3.19) で表される.

$$y(n) = a_2x(n-2) + a_1x(n-1) + a_0x(n) + b_2y(n-2) + b_1y(n-1) \quad (3.19)$$

式 (3.19) における各定数は A を式 (3.20) のように置くことにより式 (3.21) ~ 式 (3.25) によって求めることができる. ここで * は式 (3.8) により変換された後の値を表す.

$$A = 2^2 + \frac{\omega_b^*}{Q_b^*} \cdot 2T + (\omega_b^*T)^2 \quad (3.20)$$

$$a_0 = \frac{1}{A} 2T \frac{\omega_b^*}{Q_b^*} \quad (3.21)$$

$$a_1 = 0 \quad (3.22)$$

$$a_2 = -\frac{1}{A} 2T \frac{\omega_b^*}{Q_b^*} \quad (3.23)$$

$$b_1 = \frac{1}{A} \{2^3 - 2(\omega_b^*T)^2\} \quad (3.24)$$

$$b_2 = -\frac{1}{A} \{2^2 - \frac{\omega_b^*}{Q_b^*} \cdot 2T + (\omega_b^*T)^2\} \quad (3.25)$$

BPF の Q は, 中心周波数を臨界帯域の中心周波数に合わせて, 低域側の利得が-24dBのときの周波数を低域遮断周波数に合うように設定する. 低域では Notch Filter のリプルが目立つため Q を若干高くする.

サンプリング周波数が 48kHz の場合における各フィルタの特徴周波数は Bark スケールの 1/3 間隔とする. Bark スケールは蝸牛上の周波数分解能に相当する尺度として考えられている. Bark スケールと周波数の関係を示した Zwicker の近似式を式 (3.26) に示す.

$$B = 13 \arctan(0.76f) + 3.5 \arctan\left(\frac{f}{7.5}\right)^2 \quad (3.26)$$

ここで, B は Bark スケール, f は周波数を表し単位は [Hz] である. 本研究では Bark スケール 1 から 24 までを 1/3 間隔で計 72 段用いる. 最も低域である Bark スケールのフィルタ出力を 1 チャンネルと表記し, 最も高域のフィルタ出力を 72 チャンネルと表記して, 各チャンネル番号と実際の周波数の対応表を表 3.1 に示す.

表 3.1: 各チャンネル番号 (Ch. No.) に対する中心周波数の値 (サンプリング周波数=48kHz)

Ch. No.	f [kHz]	Ch. No.	f [kHz]	Ch. No.	f [kHz]	Ch. No.	f [kHz]
1	0.101	19	0.781	37	1.968	55	5.412
2	0.135	20	0.826	38	2.071	56	5.728
3	0.170	21	0.874	39	2.185	57	6.062
4	0.204	22	0.922	40	2.303	58	6.414
5	0.238	23	0.972	41	2.428	59	6.788
6	0.274	24	1.025	42	2.567	60	7.188
7	0.309	25	1.079	43	2.711	61	7.617
8	0.344	26	1.134	44	2.865	62	8.084
9	0.381	27	1.194	45	3.035	63	8.598
10	0.417	28	1.255	46	3.212	64	9.167
11	0.454	29	1.318	47	3.400	65	9.809
12	0.492	30	1.387	48	3.608	66	10.549
13	0.531	31	1.456	49	3.822	67	11.415
14	0.569	32	1.529	50	4.050	68	12.457
15	0.610	33	1.609	51	4.299	69	13.755
16	0.651	34	1.691	52	4.554	70	15.429
17	0.693	35	1.777	53	4.823	71	17.711
18	0.737	36	1.871	54	5.114	72	21.080

またサンプリング周波数が 16kHz の場合における各フィルタの特徴周波数は、1kHz~7kHz までの周波数で各中心周波数が \log スケールで等間隔になるように設定した。サンプリング周波数が 16kHz の場合の各チャンネル番号と実際の周波数の対応表を表 3.2 に示す。

3.2.2 有毛細胞モデル

有毛細胞モデルでは、蝸牛モデルで周波数分解されたデータを式(3.27)の非線型変換により整流する。

$$f(x) = \begin{cases} x^{\frac{1}{3}} & x \geq 0 \\ \frac{1}{4}x^{\frac{1}{3}} & x < 0 \end{cases} \quad (3.27)$$

また、400[Hz] 以上について-6dB/oct の特性を持つ LPF につけ、包絡線の検出を行う。本研究では高速な計算時間を要するため、FIR 型のフィルタではなく IIR 型フィルタを新たに設計した。その設計方法について以下に記す [32]。

仕様の決定 まず設計するフィルタはバターワース型 IIR-LPF であり、その周波数-振幅特性は図 3.3 で示される。

表 3.2: 各チャンネル番号 (Ch. No.) に対する中心周波数の値 (サンプリング周波数=16kHz)

Ch. No.	f [kHz]	Ch. No.	f [kHz]	Ch. No.	f [kHz]	Ch. No.	f [kHz]
1	0.100	12	0.304	23	0.926	34	2.817
2	0.111	13	0.337	24	1.024	35	3.117
3	0.123	14	0.372	25	1.133	36	3.448
4	0.135	15	0.412	26	1.254	37	3.815
5	0.150	16	0.456	27	1.387	38	4.221
6	0.166	17	0.505	28	1.535	39	4.671
7	0.183	18	0.558	29	1.698	40	5.168
8	0.203	19	0.618	30	1.879	41	5.718
9	0.225	20	0.683	31	2.079	42	6.327
10	0.249	21	0.756	32	2.301	43	7.000
11	0.275	22	0.837	33	2.546		

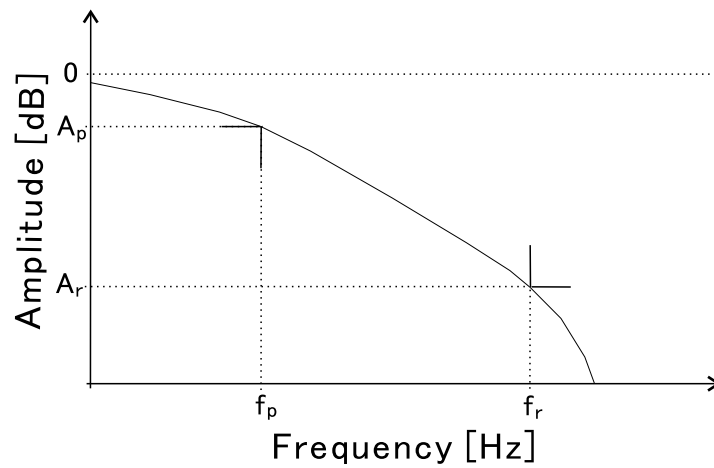


図 3.3: バタワース型フィルタの周波数-振幅特性

次数計算 次に、設定した値より必要なフィルタの次数を式 (3.28) で決定する。

$$N = \frac{\log[(10^{\frac{A_r}{10}} - 1)/(10^{\frac{A_p}{10}} - 1)]}{2 \log \frac{\omega_r}{\omega_p}} \quad (3.28)$$

この N を含む最小の整数をフィルタの次数とする。

遮断周波数の決定 遮断周波数 ω_0 は $\omega_0^p \leq \omega_0 \leq \omega_0^r$ となる値を選択すればよい。 ω_0^p , ω_0^r の値はそれぞれ式 (3.29), (3.30) より求める。

$$\omega_0^p = \frac{\omega_p}{(10^{\frac{A_p}{10}} - 1)^{\frac{1}{2N}}} \quad (3.29)$$

$$\omega_0^r = \frac{\omega_r}{(10^{\frac{A_r}{10}} - 1)^{\frac{1}{2N}}} \quad (3.30)$$

本研究では $\omega_0 = \omega_0^p$ としておく。

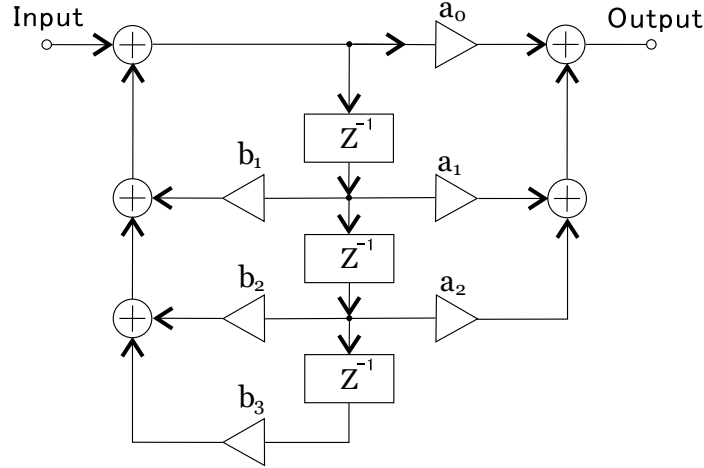


図 3.4: 3次バターワース型フィルタ

デジタルフィルタへの変換 3次のバターワース型アナログフィルタの伝達関数 $G(s)$ は式 (3.32) で表される.

$$G(s) = \frac{\omega_0^3}{(s + \omega_0)(s^2 + \omega_0 s + \omega_0^2)} \quad (3.31)$$

$$= \frac{A}{s + \omega_0} + \frac{B(s + \frac{\omega_0}{2}) + C(\frac{\sqrt{3}}{2}\omega_0)}{(s + \frac{\omega_0}{2})^2 + (\frac{\sqrt{3}}{2}\omega_0)^2} \quad (3.32)$$

式 (3.32) を z 変換を用いてデジタルフィルタへ変換すると,

$$H(z) = \frac{AT}{1 - z^{-1}e^{-\omega_0 T}} + \frac{BT(1 - z^{-1}e^{-aT}\cos(bT))}{1 - 2z^{-1}e^{-aT}\cos(bT) + z^{-2}e^{-2aT}} + \frac{CTz^{-1}e^{-aT}\sin(bT)}{1 - 2z^{-1}e^{-aT}\cos(bT) + z^{-2}e^{-2aT}} \quad (3.33)$$

となる. ただし

$$A = \omega_0, \quad B = -\omega_0, \quad C = \frac{\omega_0}{\sqrt{3}}$$

$$a = \frac{\omega_0}{2}, \quad b = \frac{\sqrt{3}}{2}\omega_0, \quad T = \frac{1}{f_s}$$

である. 式 (3.33) をまとめると,

$$H(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}}{1 - b_1 z^{-1} - b_2 z^{-2} - b_3 z^{-3}} \quad (3.34)$$

となり, 図 3.4 の回路で表すことができる.

また差分方程式に戻すと

$$y(t) = a_0 x(t) + a_1 x(t-1) + a_2 x(t-2) + a_3 x(t-3) + b_1 y(t-1) + b_2 y(t-2) + b_3 y(t-3) \quad (3.35)$$

となり，過去の時間における入出力の値よりフィルタ通過後の出力を計算することができる．ただし

$$a_0 = AT + BT$$

$$a_1 = -2ATe^{-aT} \cos(bT) - BTe^{-\omega_0 T} - BTe^{-aT} \cos(bT) + CTe^{-aT} \sin(bT)$$

$$a_2 = ATe^{-2aT} + BTe^{-\omega_0 T} e^{-aT} \cos(bT) - CTe^{-\omega_0 T} e^{-aT} \sin(bT)$$

$$a_3 = 0$$

$$b_1 = e^{-\omega_0 T} + 2e^{-aT} \cos(bT)$$

$$b_2 = -2e^{-aT} e^{-\omega_0 T} \cos(bT) - e^{-2aT}$$

$$b_3 = e^{-2aT} e^{-\omega_0 T}$$

である．

3.2.3 パルス変換部

パルス変換部は，2.2節で述べた物理的信号を電気的信号に変換する働きを持つ蝸牛神経をモデル化した蝸牛神経モデル (cochlear nerve model) を用いる．

蝸牛神経モデルの入力は有毛細胞モデルの出力であり，実数値である．この入力値から発火頻度を求め，0 または 1 の値を持つパルス列として出力する．式 (3.36) より発火間隔時間 T_i を求め，前回の発火からの経過時間がこの発火間隔時間を超えると，次の発火が起こる．ここで x は入力値， X_{max} ， X_{min} はそれぞれ入力値の最大値と最小値を表す．

$$T_i = \kappa \frac{X_{max} - X_{min}}{x - X_{min}} \quad (3.36)$$

但し入力 $x = X_{min}$ の場合発火頻度は十分大きいので任意の値とした．また，本研究では定数 κ は不応期の 10 倍程度とし，1 クロック毎に発火間隔を更新し発火の判定を行った．また実際の蝸牛神経に従って 1 個の有毛細胞あたり複数本の蝸牛神経を用意し，この複数本のパルス列で 1 つの有毛細胞出力を符号化している．

3.3 時間差検出部

時間差検出部ではパルス変換部で変換されたパルス列を図 3.5 に示す時間差検出ネットワークに入力する [30]．

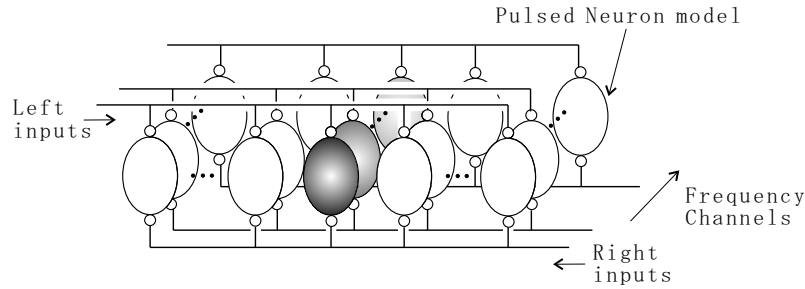


図 3.5: 時間差検出ネットワーク

本システムで用いる時間差検出ネットワークは、Jeffress のモデル [28] を基にしており、左信号を左から、右信号を右から入力する。入力されたパルス信号は単位時間ごとに矢印の方向に移動し、対応した PN モデルに入力される。各 PN モデルは二つの入力と同時に入力された時に発火する。例えば音源が中央にある場合には、左右の耳に同時に入力されるためパルス情報は同じ回数だけシフトし、中央の PN モデルにおいて同時に入力され発火することになる。音源が中央より右にある場合には、音信号は右の方が左より早く入力されるため、シフトする回数が多くなり中央より左の PN モデルにおいて同時に入力され発火する。このように音源の位置によって発火する PN モデルの場所が変わることになる。

Jeffress のモデルでは入力信号を単純にシフトしていくだけであるため、入力波形中に含まれる別の周期のパルス信号同士が同時に各 PN モデル入力されることで発火し、本来求めるべき真の時間差とは異なった偽の時間差情報が発生することになる。偽の時間差は入力信号の周期に関係するため、複数の周波数成分を比較することで真の時間差を判断することができる。しかし入力信号に多くの周波数成分が含まれてない場合には比較することが困難になるので真の時間差を判断することは難しくなる。この問題に対処するため、提案システムでは図 3.6 のように PN モデルが発火した場合、発火に関与した左右のパルス信号を以降は不必要であると判断し削除するという機構を加えた [33]。これにより偽の時間差情報の影響を小さくし真の時間差情報を明確にすることができる。

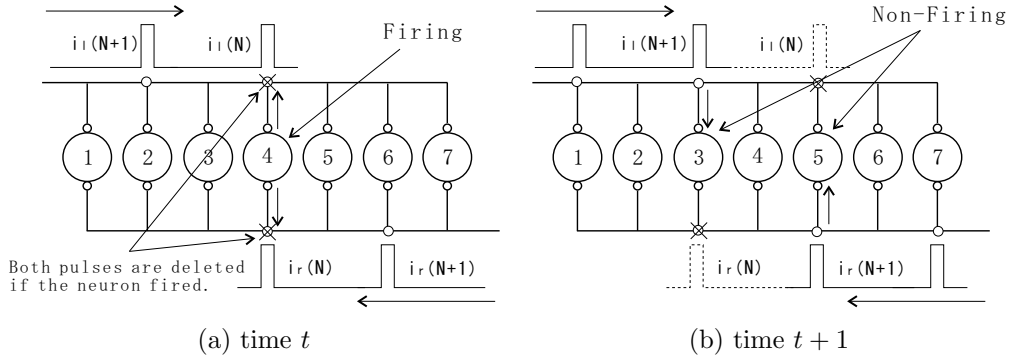


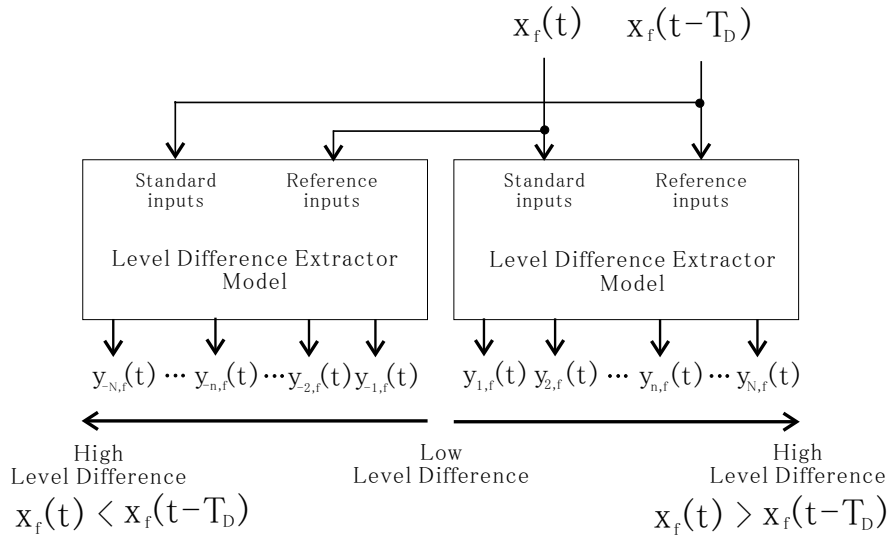
図 3.6: 時間差検出ネットワークにおけるパルス削除アルゴリズム

3.4 音圧差検出部

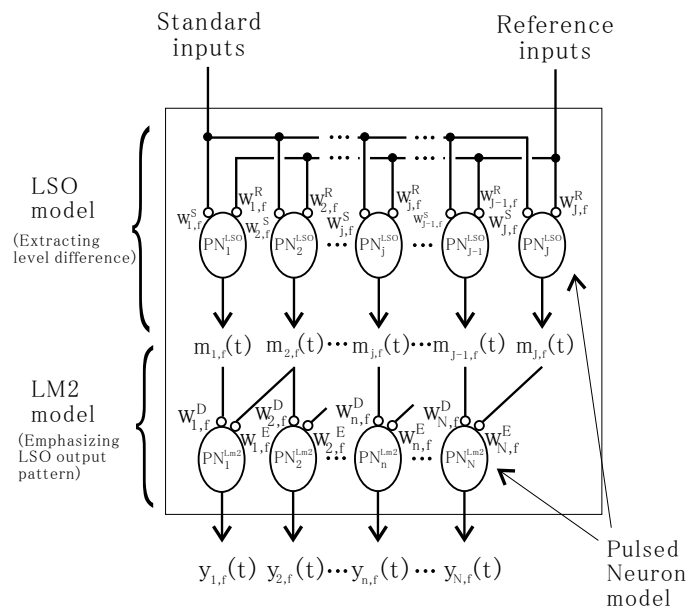
音圧差検出部では各周波数毎にパルス変換部で変換されたパルス列を図 3.7(a) に示す音圧差検出ネットワーク [30] に入力する. 音圧差検出ネットワークは, 図 3.7(b) に示すように PN モデルで構成された LSO モデルと LM2 モデルからなり, 入力された二つのパルス情報から音圧差を検出する. LSO モデルにおける各 PN モデル PN_j^{LSO} の重み $w_{j,f}^S, w_{j,f}^R$ は式 (3.37) で与えられ, 添字 n が大きいニューロンほど重みの値が負の方向に大きくなる. また入力パルスの頻度は振幅値に比例しているため二つの信号の振幅値の差が大きく, 入力パルスの頻度の差が大きくなるほど添字 n が大きいニューロンが発火するように設定されている. LM2 モデルにおける各 PN モデル PN_n^{LM2} の重み $w_{n,f}^D, w_{n,f}^E$ は式 (3.38) で与えられ, LSO モデルにおける出力パルスパターンの特徴を強調する働きがあり, 音圧差検出部の出力 $y_n(t)$ より入力信号の音圧差を判断することができる.

$$w_{j,f}^S = 1.0, w_{j,f}^R = \begin{cases} -10^{\frac{j}{\gamma}} & 0 < j < b \\ -10^{\frac{j-j}{\epsilon}} & j \geq -b \end{cases} \quad (3.37)$$

$$w_{n,f}^D = 1.0, w_{n,f}^E = -1.0 \quad (3.38)$$



(a) 全体図



(b) 音圧差検出モデル

図 3.7: 音圧差検出ネットワーク

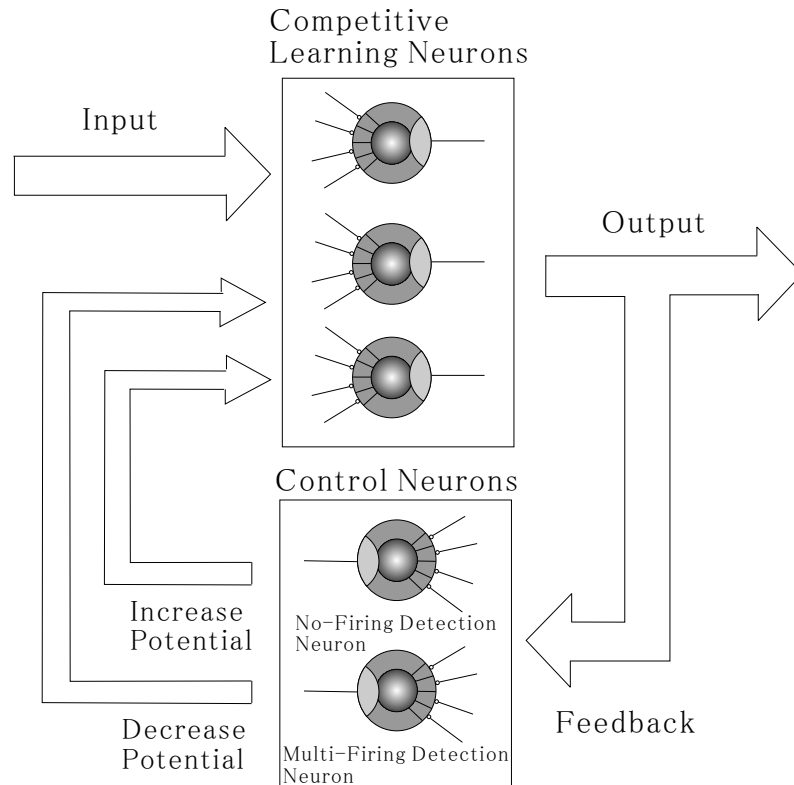


図 3.8: パルスニューロンモデルのための競合学習ネットワーク

3.5 ベクトル量子化部

周波数パターン量子化部や音圧差情報量子化部にはPNモデルによって構成された競合学習ネットワーク CONP (Competitive learning Network using Pulsed neurons)[23]を用いる。CONPの動作は自己組織化マップ (Self Organizing Map:SOM) を基にしている。SOMは、任意の次元を持つ入力信号を表現空間の次元数の削減を行ないながら位相的な関係を保存することが可能であるアルゴリズムである [34]。このため、SOMを用いることで入力信号の実際の空間における近さとSOMの素子の近さを学習によって自動的に対応づけることができる。音圧差情報量子化部のCONPでは教師なし学習を行う。この教師なし学習では、ニューロンが発火した場合に各入力 k の持つ膜電位 $p_k(t)$ の値と学習係数 α を積算した値を重みに加算した後、重みベクトルのノルムが1.0になるよう正規化して更新する。これにより各ニューロンの重みは入力ベクトルに従った値となる。そして各ニューロンは各入力ベクトルに対応することになり、ニューロンの出力からベクトル量子化を行うことができる。図3.8にCONPを示す。

CONPの学習では、入力ベクトルに対してもっとも類似した重みを持つ

ニューロン (勝者ニューロン) を決定し, 勝者ニューロンのみが発火して学習するようにしなければならない. しかし出力値が実数値ではなく 0 または 1 のパルス信号であり各素子が独立非同期に動作している PN モデルでは勝者を決定することが困難である. そこで, 図 3.8 のように外部の素子 (制御ニューロン) が出力の状態を判断し各競合学習ニューロン (Competitive Learning Neuron:以下 CL ニューロン) のもつ内部電位を一律に調整し, CL ニューロンが一つだけ発火するような状態にするという制御を行なっている [23]. 制御ニューロンとしては, CL ニューロンが1つも発火していないことを検出する無発火検出 (No Firing Detection) ニューロンと, CL ニューロンが複数個発火していることを検出する複数発火検出 (Multi Firing Detection) ニューロンを用意する.

CONP においては, 各 CL ニューロンの閾値は外部素子を用いて増減されるため, 指定値以下の入力に対して発火しないようにするという従来のニューロンモデルにある閾値制御はできない. このため各 CL ニューロンには学習により変化する重みの他に, 常に 1.0 とする重みとそれに関する電位を持たせる. 重みが固定値であるためこれによって計算される電位は入力パルスの頻度を示すことになる. そして閾値 θ の他に制御閾値 θ_{gate} を設定することで, 入力パルス頻度が制御閾値以下であれば発火しないという制御を行う. また入力パルス頻度が制御閾値以下であれば制御ニューロンからの閾値の調整は行わない. 各 CL ニューロンにおいては内部電位から入力パルス頻度を一定の割合で引くことで, 内部電位の急激な変化を抑え, CL ニューロンの内部電位の調整における追従性を向上させている.

CONP における内部電位の調整を考慮すると各 CL ニューロンにおける出力の式 (3.3) は, 式 (3.39) のようになる. ここで $p_{nfd}(t)$ は無発火検出ニューロンに対する膜電位, $p_{mfd}(t)$ は複数発火検出ニューロンに対する膜電位, $p_{in}(t)$ は入力パルス頻度, β は入力パルス頻度に対する割合である ($0 \leq \beta \leq 1$).

$$o(t) = f_{usf} \left(\sum_{k=1}^K p_k(t) - \theta + p_{nfd}(t) - p_{mfd}(t) - \beta \cdot p_{in}(t) \right) \quad (3.39)$$

3.6 パターン識別部

音源種類及び接近識別部では音圧差情報量子化部でベクトル量子化されたパターンが入力され、入力パターンからどの音であるかを識別すると同時に、その音源が接近しているか離脱しているかということも識別する。音源種類及び接近識別部も CONP で構成されている。出力部として競合学習ネットワークを用いた理由として、背景雑音やなどの情報に対して頑健にする他に、適用する環境で学習させることによって識別するためのパラメータである各ニューロンの重みの状態をシステムが自動的に環境にあわせることができ利用者が手動で設定するという不便さをなくすという目的がある。

学習に関しては、識別すべき音の種類を明示的に示す必要があるため黒柳らの提案した PN モデルにおける教師あり学習法 [35] に基づき計算機上で各 PN モデルの重みを更新する。具体的には、まず学習のために出力に関する膜電位 $p_{out}(t)$ と教師パルスに関する膜電位 $p_{sup}(t)$ を用意しておく。各ニューロンが発火し、かつ教師パルスが与えられた場合に式 (3.40) に従い重みが更新される。この教師あり学習により各ニューロンはそれぞれが識別すべき音の入力にもっとも適合することになり、ニューロンの出力と識別音を対応させることができる。

$$w_k(t + \Delta t) = w_k(t) + \alpha \cdot p_{in_k}(t) \cdot \{p_{sup}(t) - p_{out}(t)\} \quad (3.40)$$

ただし制御ニューロンからの閾値制御により、必ずいずれかのニューロンが出力結果を返すことになってしまうため、後述の実験においては学習した目的音以外の音が入力された場合のために“その他”と識別するための素子を用意した。

第4章

聴覚情報処理システムのハードウェア実装

システムの実用化を考える上で、利用者へ警告するための出力するまでの処理速度は非常に重要な要因である。一般にパターン認識を行う場合には複雑な計算を必要とするため処理時間は膨大なものとなる。これに対処するため、専用のCPUを用意したりハードウェアに実装することで高速化を行うことが多い。前章でも説明した通りPNモデルの利点として簡単な電子回路でハードウェア実装が可能であり、回路規模を小さくすることができることがあげられる。回路規模を小さくできれば消費電力やコストの面で優位となる。本研究ではハードウェア実装にはField Programmable Gate Array(FPGA)を使用する。本章ではこのFPGAについて述べた後、PNモデルをFPGAを用いて実装する方法について説明する。

4.1 FPGA(Field Programmable Gate Array)

FPGAはユーザの手によって内部の論理回路を自在に変更することができるProgrammable Logic Device(PLD)の一種である。最初にFPGAが登場してからの20年程度の間にはプログラマブルデバイスの高集積化、多機能化、高速化、低価格化が進んできた[22]。主な用途としてはプログラムによる回路の書き直しが何度でも行える点から回路設計などに利用される。一般に多数の集積回路を製造する場合にはASIC(Application Specific Integrated Circuit)を利用することが多く、同じ回路規模で比較すると、FPGAの製造コストはASICと比べて高くなる。しかしASICは開発にかかる技術資産や設備などが高価でありその設計資産を他に流用することが困難であるため少量多品種の開発においてはFPGAが優位である。

FPGAは図4.1のようにプログラム可能な比較的小規模である論理ブロッ

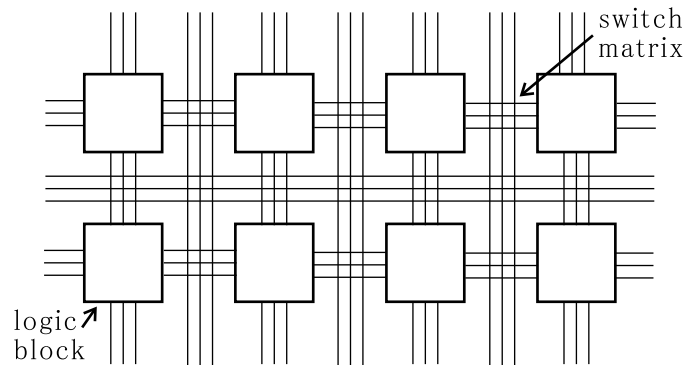


図 4.1: FPGA の基本構造

クを格子状に配置し，その間に配線路を設けた構造を基本としている．一つの論理ブロック自体は小規模であるが，多くのブロックを組み合わせることで大規模な回路を実現することができる．論理ブロックの基本要素はSRAMで構成されたLUT(Look-Up Table)，D-FF，ルーティング資源である．SRAMでは電源を切ると回路情報が揮発してしまうため，動作させる際に外部の専用ROMに接続し回路情報を読み込んでから(コンフィグレーション・データ)使用する形をとる．また，回路情報の保持要素としてEEPROMを用いるFPGAもある．EEPROMは電源を切っても回路情報を保持しているため，電源投入後すぐに利用可能である．

論理ブロックは配線路に設けられたスイッチ・マトリクスによって任意の接続が可能である．スイッチ・マトリクスはトランスファ・ゲートで作られており，そのON/OFFを決定するのもSRAMで構成されている．

FPGAの開発においてFPGAに回路を書き込むためのプログラム言語としては，ハードウェア記述言語(Hardware Description Language:HDL)が使われる．現在，HDLはVerilog HDL[39]とVHDL[40]の二つが主流である．FPGAが普及し始めた当初はVHDLにしか対応していない開発ツールが多かったためVHDLが広く利用されていたが，近年ではVerilogHDLが利用されるようになってきており，さらにSystemC[41]と呼ばれるC言語のアルゴリズム記述から回路合成を行う技術も発達してきている．本研究ではVHDLを使用して回路を記述した．

4.2 パルスニューロンモデルの実装

従来のニューロンモデルにおいては入力信号がアナログ値であるため、“入力×重み”という計算を行うために積算器を必要とする。しかしPNモデルでは入力が0か1のパルス信号であるため、“入力×重み”の計算が積算ではなく、入力信号が入力された時に重みの値だけ出力するAND回路で実現することができる。一般にAND回路は積算器に比べて回路規模が小さく、さらに各回路は入力の本数だけ必要となるためPNモデルは従来のニューロンモデルに比べて非常に小さい回路規模で実装することができる。AND回路より出力されたそれぞれの値は加算器を用いて加算され、比較器にて閾値と比較して出力が決定される。

ただし本研究においては膜電位の時定数が全て同じ値であり時間によって変化しないと仮定する。これによって式(3.2)、式(3.3)で示される内部電位 $I(t)$ の計算は

$$\begin{aligned}
 I(t) &= \sum_{k=1}^K p_k(t) \\
 &= \sum_{k=1}^K \{p_k(t - \Delta t) \exp(-\frac{\Delta t}{\tau}) + w_k \cdot i_k(t)\} \\
 &= C_t \sum_{k=1}^K p_k(t - \Delta t) + \sum_{k=1}^K w_k \cdot i_k(t) \\
 &= C_t I(t - \Delta t) + \sum_{k=1}^K w_k \cdot i_k(t) \\
 &= I(t - \Delta t) + \sum_{k=1}^K w_k \cdot i_k(t) - \delta I(t - \Delta t) \quad (4.1)
 \end{aligned}$$

とできる。ただし $\exp(-\frac{\Delta t}{\tau}) = C_t = (1 - \delta)$ である。

式(4.1)のうち、第一項は前時間からの内部電位の値を示し、レジスタに記憶させておく。第二項は重み w_k と入力パルス $i_k(t)$ のAND演算をとりその総和を求めることで実現できる。第三項は時定数による減衰を示しており、ビットシフトと補数演算で実現しており元の内部電位の何分の一かを減算するという演算を行う。本来減衰は各膜電位毎に用意するものであるが、本実験では各膜電位の時定数を同じ値としたため膜電位の総和である内部電位に対し減衰をもたせることで等価な回路とした。図4.2のように減衰において N ビットシフトを行う場合、電位の値は $2^N - 1$ より小さくなら

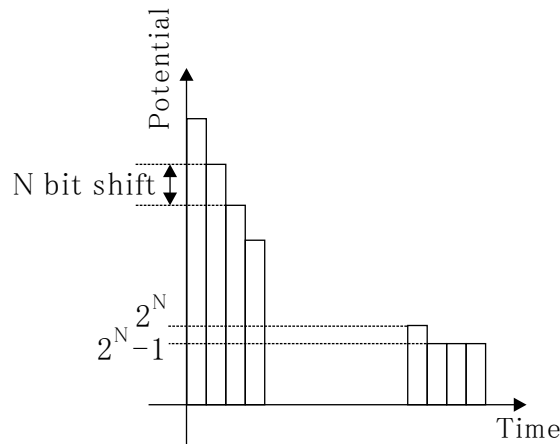


図 4.2: FPGA における電位の減衰の表現

ず、0にはならない。このため、各重みや閾値を設定する際に電位の基準値を $2^N - 1$ に設定し、基準値に対して重みや閾値が何倍であるかを算出してから与えるという方法を用いた [17].

以上の構成により FPGA 上で電子回路として実現した PN モデルのブロック図を図 4.3 に示す。

なお今回の実装にあたっては、PN モデルの持つ学習機構は FPGA 上には実装されていない。このため、計算機上であらかじめ学習して生成された重みを固定小数点表現に変換して回路に与えることで代用し、認識動作のみを行うこととしている。

4.3 時間差検出部の実装

時間差検出ネットワークを実装するには、図 4.4 に示すようにパルス情報を移動させるためのシフト演算器と PN モデルを横一列に配置する。動作については 3.3 節で説明した通りである。また改良点である PN モデルが発火した場合にパルスを削除するという機構については、各 PN モデルの出力を対応したシフト演算器へと送り、PN モデルの出力があった場合にシフト演算器の入力をクリアするという方法を用いている。

4.4 音圧差検出部の実装

PN モデルによる音圧差検出部のうち、LSO モデルに関しては FPGA 上では図 4.5 のように構成した。本来図 4.3 のように各ニューロン毎に各入力に

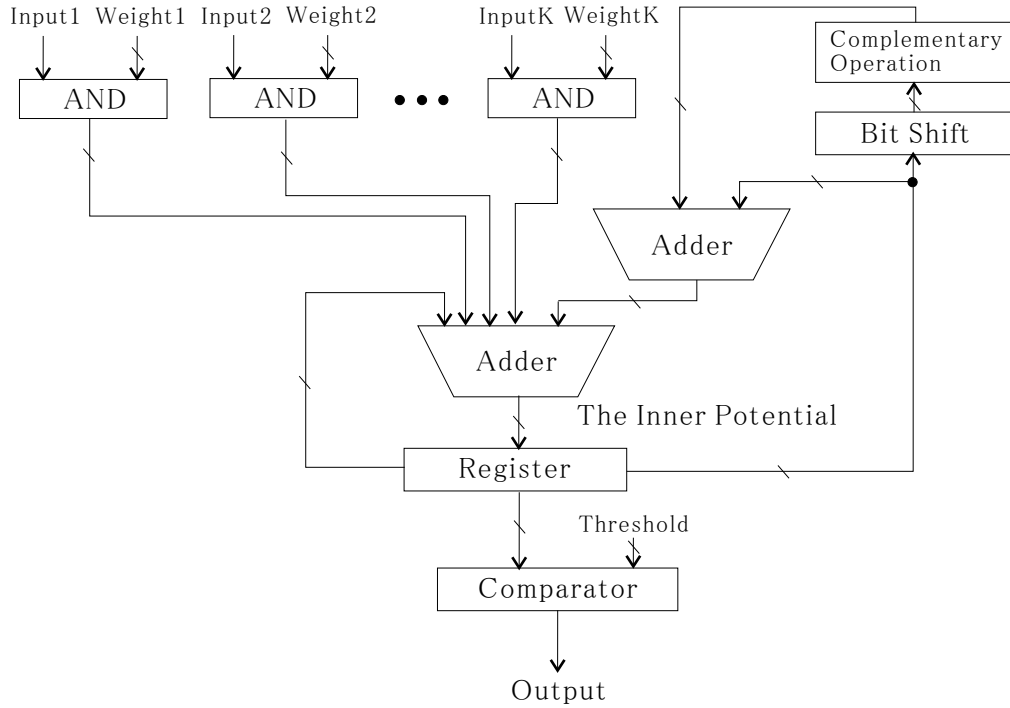


図 4.3: FPGA におけるパルスニューロンモデル

対する重みである AND 素子を用意することが適切であるが、図 3.7(b) に示す重み $w_{j,f}^S$ は式 3.37 に示すように 1.0 で一律でありかつ入力も全てのニューロンに同時に与えられるため一つの AND 回路を用いるだけで等価となる。

LM2 モデルに関してはそれぞれ $w_{n,f}^D, w_{n,f}^E$ である 2 つの重みの AND 回路を持つ PN モデルを複数個並べることで構成した。

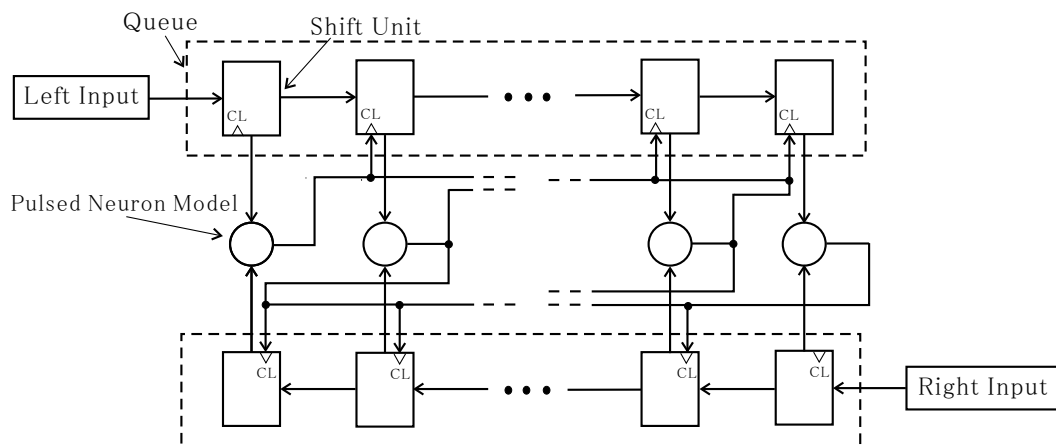


図 4.4: FPGA における時間差検出ネットワーク

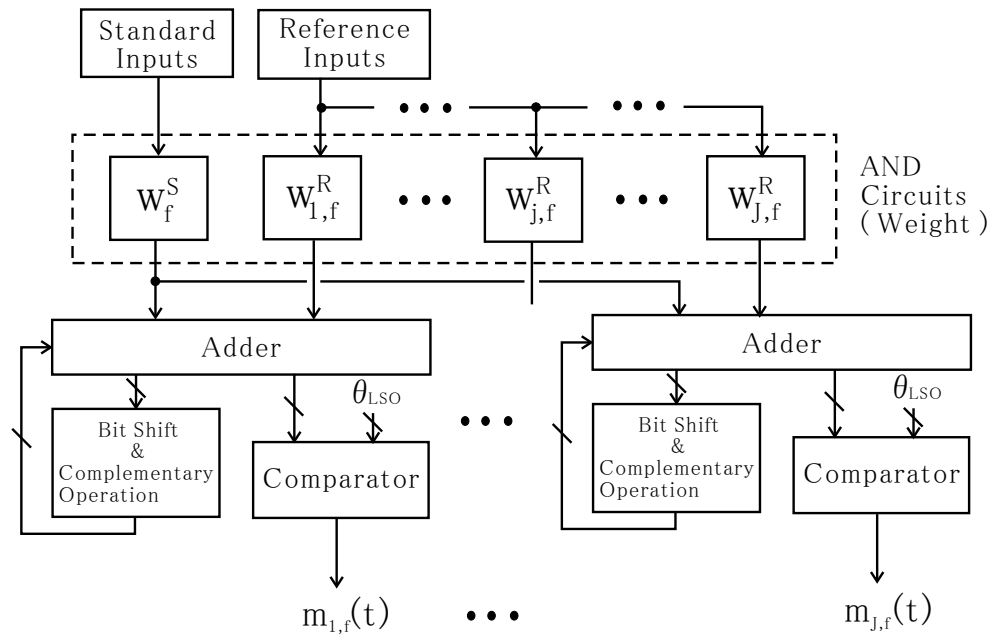


図 4.5: FPGA における LSO モデル

4.5 競合学習ネットワークの実装

PN モデルによる競合学習ネットワークは図 4.6 のように実装した。図 4.3 の FPGA による PN モデルを基本とし、図 4.7 の競合学習ニューロン (以下 CL ニューロン) と図 4.8 制御ニューロンを配置する。入力パルスの頻度の計算については全ての CL ニューロンにおいて時定数及び重みは同じであるため、図 4.9 に示す入力パルス頻度を計算するためのニューロン (Input Potential Control Neuron: 以下 IP ニューロン) を外部に 1 つ用意することで等価な回路とした。

そして単一発火のための制御方法は、内部電位を調整することは閾値を調整することと等価であるため、各 CL ニューロンの内部電位に対して直接加減算するのではなく外部から閾値を与えて発火の判定を行うことで実現した。即ち入力パルスの頻度から内部電位の減少させるには IP ニューロンから CL ニューロンに対して閾値 $\theta_{in}(t)$ を与えることで等価な回路とした。同じく制御ニューロンからの閾値の制御に関しても外部に図 4.10 のようなユニット (Threshold Control Unit: 以下 TC ユニット) を無発火検出用と複数発火検出用に 1 つずつ用意して閾値 $\theta_{tc}(t)$ ($\theta_{nfd}(t)$ と $\theta_{mfd}(t)$ の 2 種類がある) を CL ニューロンに与えることで等価な回路とした。以上より、FPGA に実装した場合の各 CL ニューロンにおける出力は式 (4.2) のようになる。これ

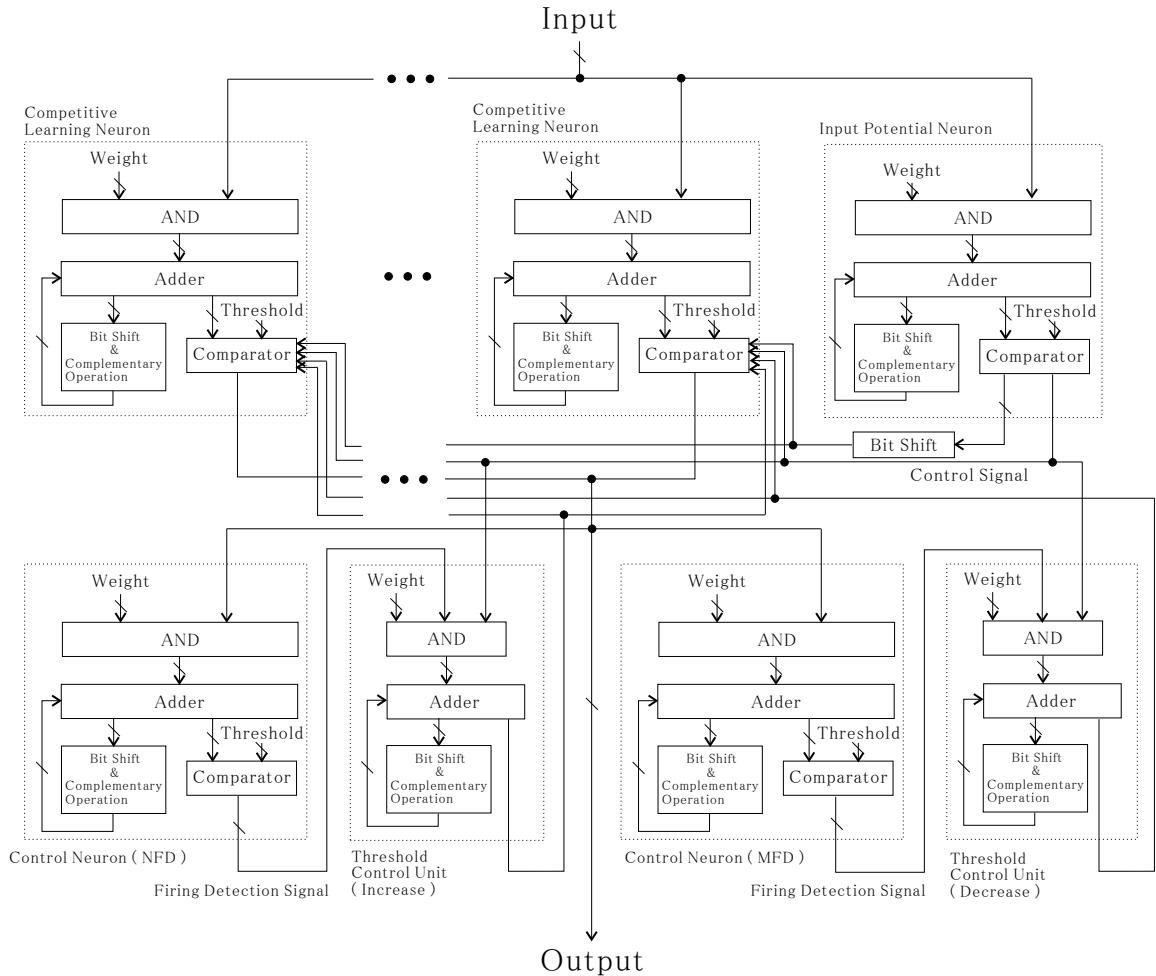


図 4.6: FPGA における競合学習ネットワーク

は式 (3.39) と等価である.

$$o(t) = f_{usf} \left(\sum_{k=1}^K p_k(t) - \theta + \theta_{nfd}(t) - \theta_{mfd}(t) - \theta_{in}(t) \right) \quad (4.2)$$

続いて動作について説明する. まず入力パルスは各 CL ニューロンと IP ニューロンに入力され各ニューロンが動作する. CL ニューロンの出力は2つの制御ニューロンに送られ各制御ニューロンにて CL ニューロンが一つも発火していないもしくは複数個発火していることを検出して出力する (図 4.6 中 Firing Detection Signal). 一方, IP ニューロンの出力は CL ニューロンの比較部に送られ, IP ニューロンの出力が 0 の場合には入力パルス頻度が充分でないとして CL ニューロンは発火しないようにする (図 4.6 中 Control Signal). また, 各制御ニューロンの出力は TC ユニットに送られる. TC ユニットにおいては, Control Signal が出力されているときのみ, Firing

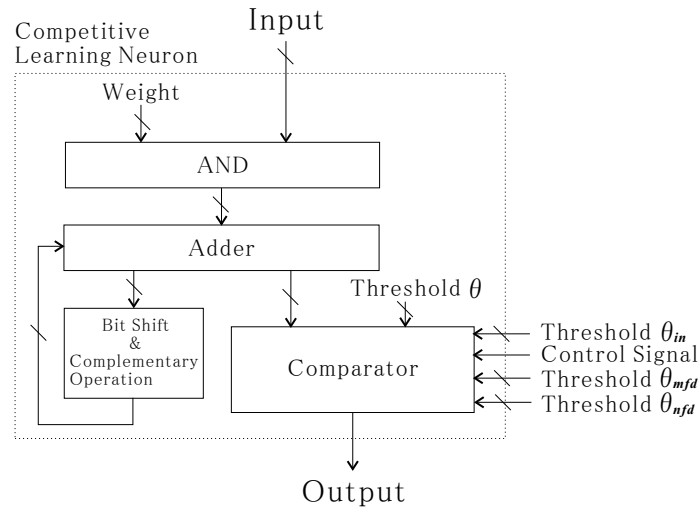


図 4.7: FPGA における競合学習ニューロン

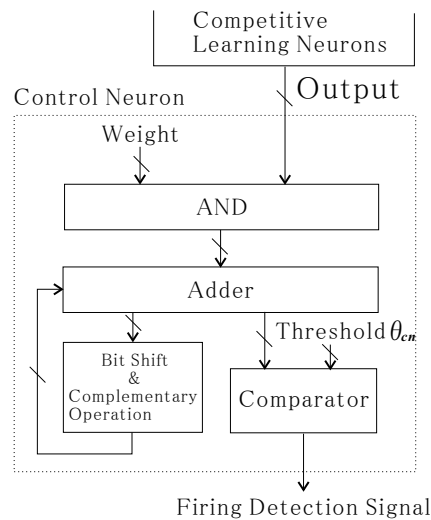


図 4.8: FPGA における制御ニューロン

Detection Signal が送られた際に TC ユニット中の閾値 θ_{tc} が加算されるようになっている。この閾値は各 CL ニューロンに与えられ CL ニューロンの閾値が調整される。IP ニューロンの出力には上記の Control Signal の他に閾値があり、入力パルス頻度である IP ニューロンの電位より一定の割合 β を与えた閾値 θ_{in} が各 CL ニューロンに与えられる。これら 2 つの閾値調整 (図 4.6 中 Threshold Control) により各 CL ニューロンのうち 1 つだけが発火できるよう制御される。

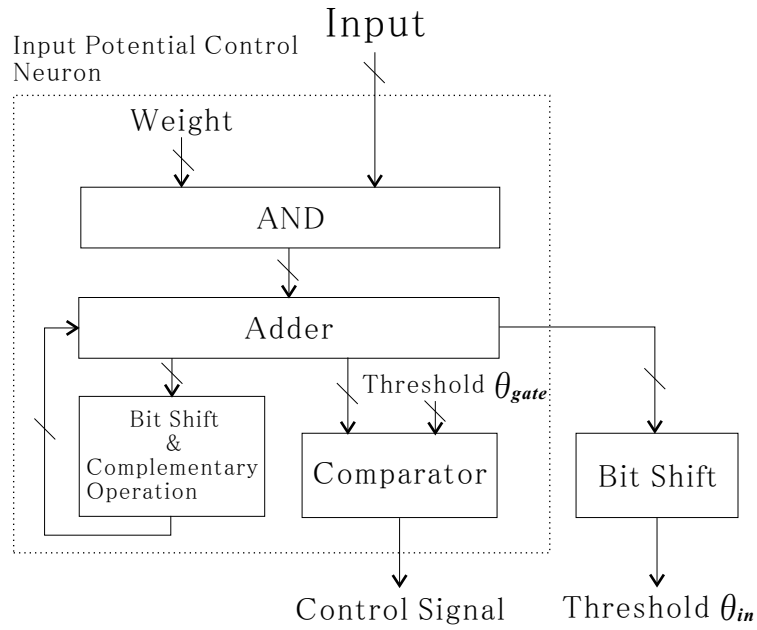


図 4.9: FPGA における入力パルス頻度を計算するニューロン

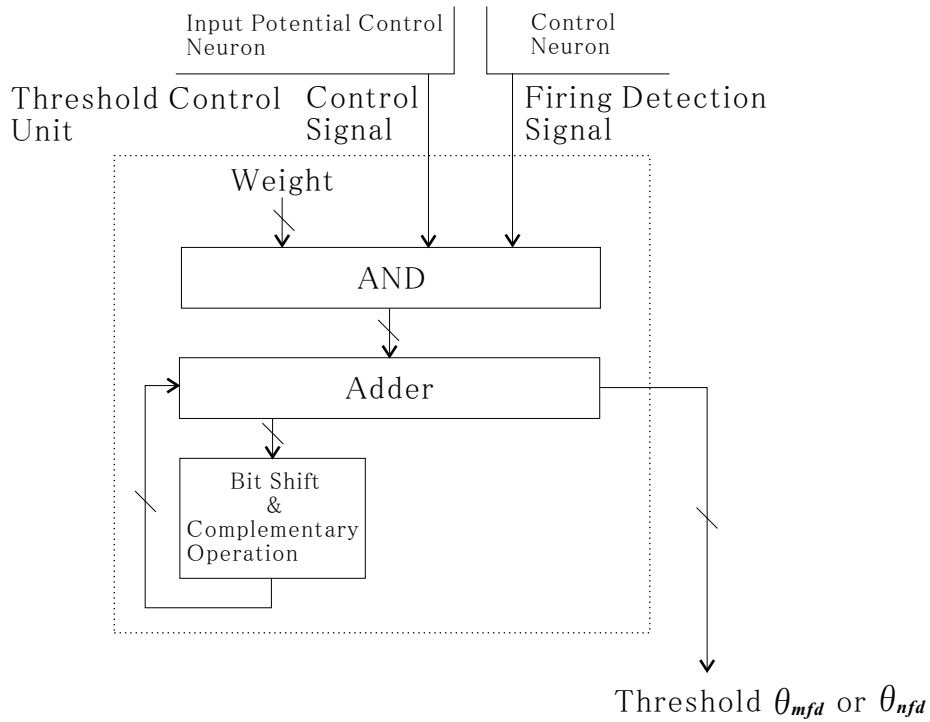


図 4.10: FPGA における閾値制御を行うユニット

第5章

音源定位及び音源種類識別実験

本章では、前章までに説明した聴覚情報処理システムの一つとして、音源定位と音源種類の識別を行うシステムを構成する。また各種実験によりFPGAを用いてPNモデルをハードウェア実装した本システムが入力音から音源定位と音源種類の識別を行うことができ、他手法より少ない回路規模で実現できることを示す。

5.1 システム構成

本論文で構成する音源定位及び音源種類識別を行う聴覚情報処理システムについて概略図を図5.1に示す。本システムは大きく分けて周波数-パルス変換部(図5.1中A)、時間差検出部(B-1)、音源位置識別部(B-2)、周波数パターン量子化部(C-1)、音源種類識別部(C-2)の5つからなる。時間差検出部、音源位置識別部、周波数パターン量子化部、音源種類識別部についてはPNモデルで構成されFPGA上に実装されている。入力された音信号は計算機上の周波数-パルス変換部(A)で複数の周波数成分に分割された後パルス情報へと変換される。変換されたパルス情報はFPGAへ入力され音源定位及び音源種類の識別が並列に行われる。得られた音源定位結果及び音源種類の識別結果は再び計算機へ返され、計算機上で出力結果を表示する。

音源定位手法としては、黒柳らの提案した時間差検出手法[30]を用い、時間差検出部(B-1)にて入力される音信号に対する左右の信号の時間差情報を検出する。次に、検出された時間差情報を音源位置識別部(B-2)である競合学習ネットワークに入力することで音源位置の識別を行う。出力部として競合学習ネットワークを用いた理由として、雑音などの誤った情報に対して頑健にする他に、適用する環境で学習させることによって教師となる信号を与えることなくシステムが自動的に識別するためのパラメータである各

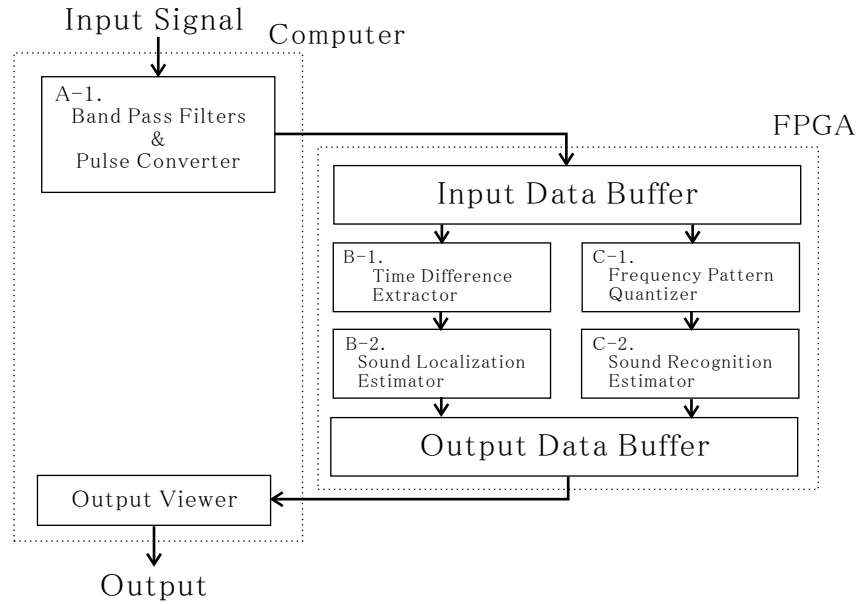


図 5.1: 音源定位及び音源種類識別を行うシステムの概略

ニューロンの重みの状態を環境にあわせ利用者が手動で設定するという不便さをなくすという目的がある。

音源種類の識別手法としては、周波数-パルス変換部で得られた周波数パターンを用いる。しかしこの周波数パターンを用いて直接教師あり学習を行うと同じ音の周波数パターンでも時間的に周波数成分が変化する場合に学習が難しくなるという問題がある。そこで本システムでは周波数パターンを一度周波数パターン量子化部 (C-1) である競合学習ネットワークによってベクトル量子化し、ベクトル量子化されたパターンを後段の音源種類識別部 (C-2) である競合学習ネットワークに入力することで入力音源の種類を識別する。

5.2 実験条件

本システムの実装に用いた計算機及びFPGAの仕様を表5.1に示す。録音用マイクにはダミーヘッド KU100 を用いた。また音源は図5.2のようにダミーヘッドの左耳に対して垂直に約1m離れた位置を開始点 (left 90°) とし、ダミーヘッドを中心とした円状に右耳まで (right 90°) 30度ずつ移動させた。学習用の音には残響などの影響がない無響室で録音した音を用い、認識実験用の音には実環境で動作することを想定し広さ5.6m×6.0m、コンクリート壁とガラス窓のある室内で録音した音を用いた。

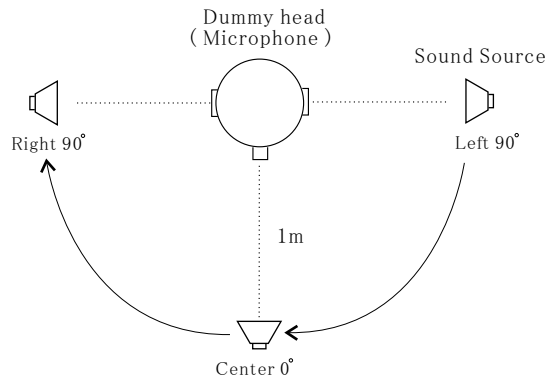


図 5.2: 音源定位実験における音源の移動

実験に用いた入力音源の種類としては、音源定位の学習のために全ての周波数成分を持つ信号であるホワイトノイズと、音源種類の識別のための学習音に“警報ベル”，“インターホン”，“やかんでの水の沸騰音”，“ホワイトノイズ”，“黒電話の呼び出し音”，“音声(成人男性の母音「あ」)”の合計6種類を用意した。

定位の解像度としては前方向30度ずつ合計7方向としている。ただし音源位置識別部では教師なし学習が行われているため各出力素子が正確な角度を示しているわけではない。音源位置識別部の出力素子を必要に応じて増やせば解像度を上げることも可能であるが[23]，本システムの応用先として聴覚障害者支援や室内での警告装置を考えており，細かな定位精度は必要なく大まかな方向さえ定位できればよいため，7方向という解像度でも問題なく，正確な角度は必要ないと判断した。

また音源種類の識別用に用いた学習音について，周波数-パルス変換部の出力を図5.3に示す。図5.3の横軸は時間，縦軸は周波数帯域番号であり上にあるほど高い周波数帯域である。各ます目における濃淡は出力パルス信号の頻度を示しており，黒いほど多くのパルス信号が出力されたことを示す。なお“雑踏”に関してはその他に分類させる目的で用意してある。初めにこれらの音を用いて表5.2のパラメータに従い計算機上で各CONPの学習を行った。その後，学習の結果得られた重みやパラメータを用いFPGA上に各部を実装した。実装の際に必要な各部の回路数を表5.3にあげる。必要となる回路数の合計が30,291[ALUTs]であるため，本実験で用いた，本研究室で設計・開発した総回路数が48,352[ALUTs]であるFPGAボード(図5.4)に全ての回路を搭載することが可能である。

表 5.1: 使用する機器の仕様

計算機		FPGA	
CPU	Pentium4 3.8GHz	開発ボード	Altera 社 StratixII EPS60
メモリ	2 GB	総回路数	48,352[ALUTs]
OS	WindowsXP	計算機との 通信規格	USB 2.0
コンパイラ	Microsoft Visual Studio C++.NET	論理合成 ソフトウェア	Altera 社 QuartusII 5.0

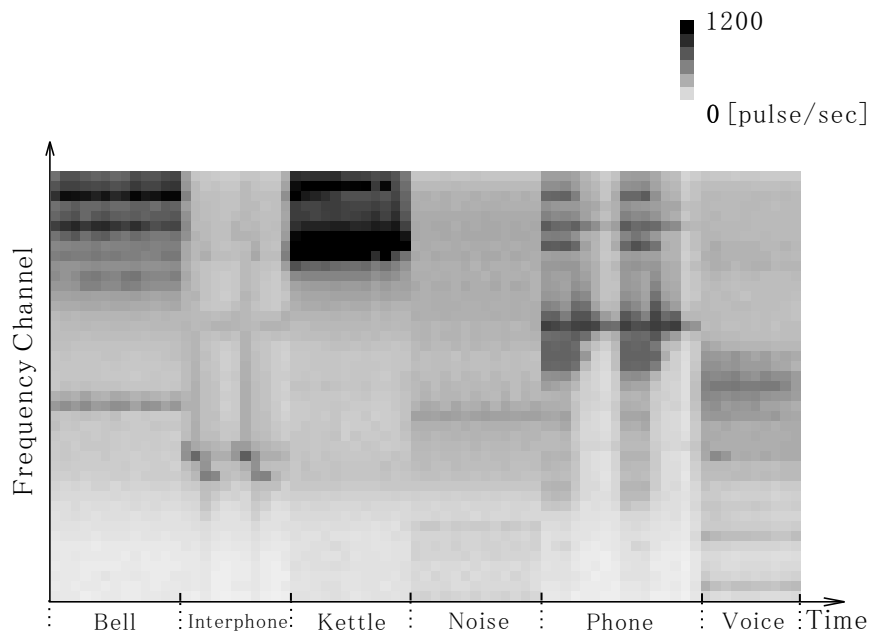


図 5.3: 音源種類識別実験のための学習音

以降の節では、学習音とは別に録音した音を使用して実験を行った際の各出力結果について検討していく。

表 5.2: 音源定位及び音源種類識別実験におけるパラメータ

入力音	
サンプリング周波数	16[kHz]
量子化ビット数	8[bit]
周波数-パルス変換部	
音源定位に用いる周波数帯域数	15[Channel]
音源種類の識別に用いる周波数帯域数	43[Channel]
時間差検出部	
総シフト回数	41[回]
出力ニューロン数	21[個]
閾値 θ_{mso}	1.0
時定数 τ_{mso}	0.35[ms]
不応期 RP_{mso}	1.0[ms]
競合学習ニューロン	
音源位置識別部 CL ニューロン数	7[個]
周波数パターン量子化部 CL ニューロン数	10[個]
音源種類識別部 CL ニューロン数	6[個]
閾値 θ	1.0×10^{-4}
制御閾値 θ_{gate}	150.0
入力パルス頻度からの減算率 β	3.125×10^{-2}
膜電位減衰の時定数 τ_p	20[ms]
不応期 RP_{CL}	1.0[ms]
無発火検出ニューロン / 複数発火検出ニューロン	
時定数 τ_{no} / τ_{multi}	0.5 / 1.0[ms]
閾値 $\theta_{no} / \theta_{multi}$	$-1.0 \times 10^{-3} / 2.0$
CL ニューロンへの結合重み	0.5 / -0.5
競合学習パラメータ	
学習係数 α	2.0×10^{-7}
学習回数	1000
学習半径の収束回数	800

表 5.3: FPGA 上での各回路の回路数

部位	回路数 [ALUTs]
FPGA 入出力部	83
時間差検出部	13,621
音源位置識別部	7,478
周波数パターン分類部	3,923
音源種類識別部	1,416
合計	26,521

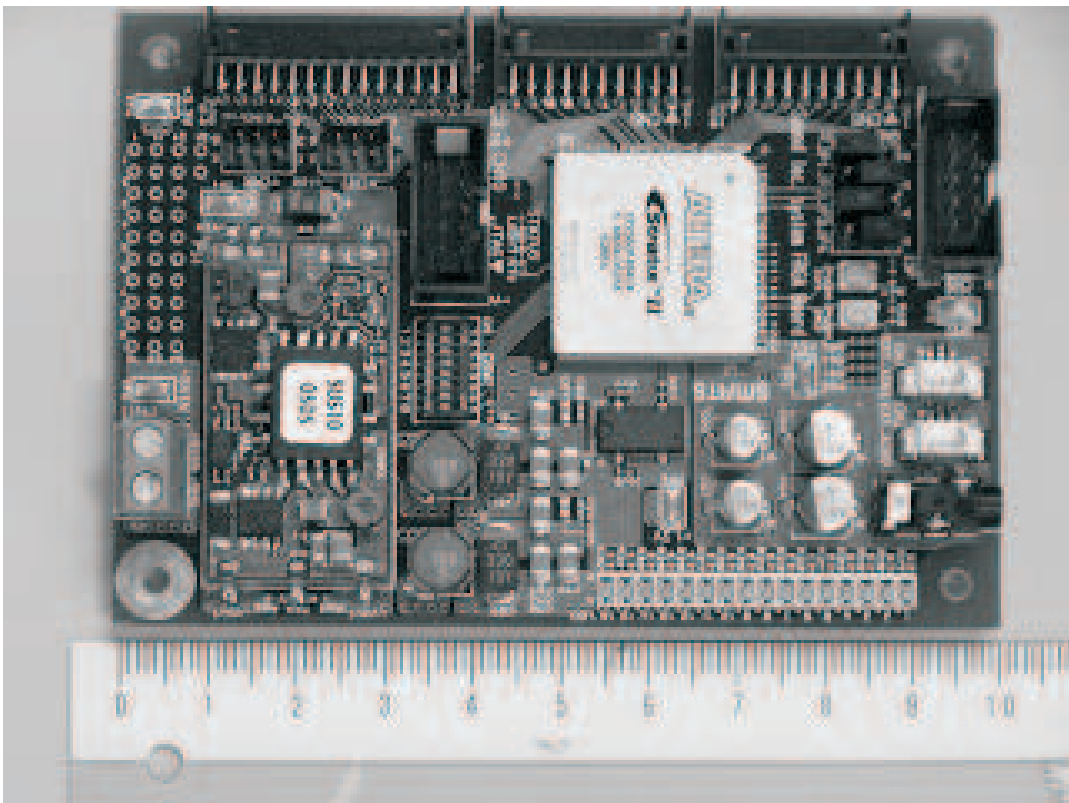


図 5.4: 本研究で使用了 FPGA ボード

5.3 時間差検出結果

音源位置の定位を行うにはまず時間差検出部において左右の音信号の時間差を検出する。図 5.5 に音源が正面 (center 0°) にある場合の時間差検出部の出力結果を示す。

図 5.5 における横軸は時間差検出部における PN モデルのニューロン番号、縦軸は周波数帯番号であり上にいくほど高い周波数帯域であることを示す。各ます目における濃淡は出力パルス信号の頻度を示しており、黒いほど多くのパルス信号が出力されたことを示す。図 5.5 の結果より、ほぼ全ての周波数帯域において中心部分が黒くなっている、これは時間差が 0 であることを示す PN モデルの出力パルス頻度が高いことを表している。すなわち音源が中央にあることを示しており、正しく時間差が検出できているといえる。また計算機と FPGA の結果を比較すると出力しているニューロンに大きな違いはなく、FPGA による時間差検出部でも正しく時間差を検出できている。ただし警報ベルのように特定の周波数成分のみを持つ信号の場合、時間差を検出できる周波数も特定の周波数領域に限られることになり時間差から音源の位置を推定することが難しくなる。

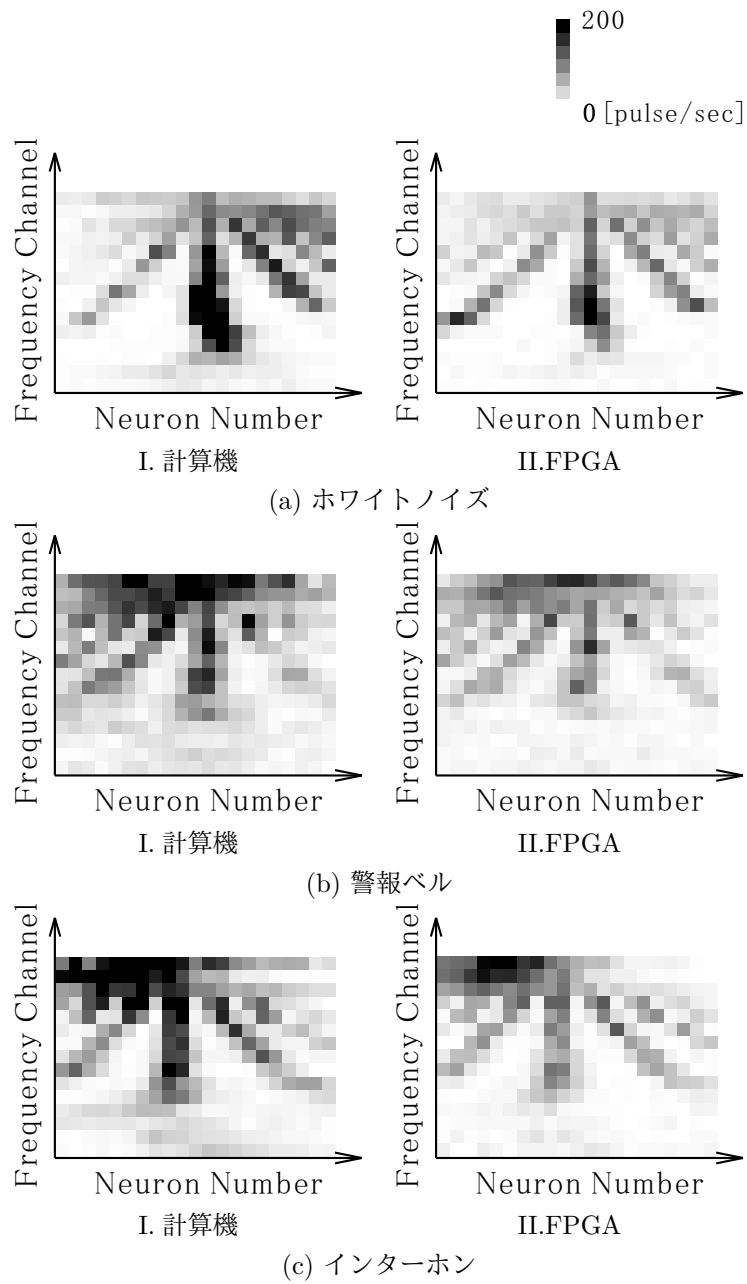


図 5.5: 時間差検出部の出力

表 5.4: 各入力音に対する音源位置識別部の出力結果

(a) white noise							
Input Location	Output Rate [%]						
	left90°	left60°	left30°	center0°	right30°	right60°	right90°
left 90°	<u>98.3</u>	0.4	0.3	0.3	0.3	0.2	0.2
left 60°	0.4	<u>98.1</u>	0.3	0.3	0.3	0.3	0.3
left 30°	0.3	0.3	<u>98.4</u>	0.3	0.3	0.2	0.2
center 0°	0.3	0.2	0.3	<u>98.6</u>	0.2	0.2	0.2
right 30°	0.2	0.2	0.2	0.3	<u>98.5</u>	0.3	0.3
right 60°	0.3	0.3	0.3	0.3	0.3	<u>98.2</u>	0.3
right 90°	0.2	0.2	0.3	0.3	0.3	0.3	<u>98.4</u>

(b) alarm bell							
Input Location	Output Rate [%]						
	left90°	left60°	left30°	center0°	right30°	right60°	right90°
left 90°	<u>61.3</u>	15.5	8.9	3.5	4.6	3.0	3.1
left 60°	19.7	<u>47.6</u>	7.1	2.0	14.4	5.2	4.1
left 30°	10.6	4.0	<u>69.8</u>	1.7	2.6	8.0	3.3
center 0°	3.9	2.6	2.2	<u>80.9</u>	1.4	3.8	5.2
right 30°	4.4	9.9	3.0	1.6	<u>56.9</u>	5.4	18.9
right 60°	4.8	6.3	13.5	3.7	6.1	<u>46.7</u>	18.9
right 90°	2.1	3.2	5.9	2.6	8.3	10.9	<u>66.9</u>

5.4 音源定位実験

続いて、時間差検出部で得られた時間差情報を音源位置識別部に入力することで音源の定位を行う。表 5.4 にそれぞれの位置に音源がある場合の音源位置識別部における各ニューロンの出力の割合を示す。表中の数字は各入力音を与えた場合における各出力ニューロンの単位時間あたりの出力パルス数 ([pulse/s]) の割合である。また最大頻度となる出力結果に下線が引いてある。結果よりそれぞれ角度にあわせて対応したニューロンが最も多く出力しており正しく定位ができていることがわかる。入力音が警報ベルの場合ではホワイトノイズの場合に比べて対応したニューロンの発火の割合は小さくなっている。この理由は図 5.5 に示すように音源の種類によって特定の周波数帯域のみ時間差が検出できることになり音源の位置を特定することが困難になるためである。

また図 5.6 に音源を左 90 度から右 90 度まで連続して移動させた場合のホワイトノイズと音声における音源位置識別部の出力結果の時間的变化を示す。図 5.6 における横軸は時間を示し、縦軸は CONP の出力素子番号であ

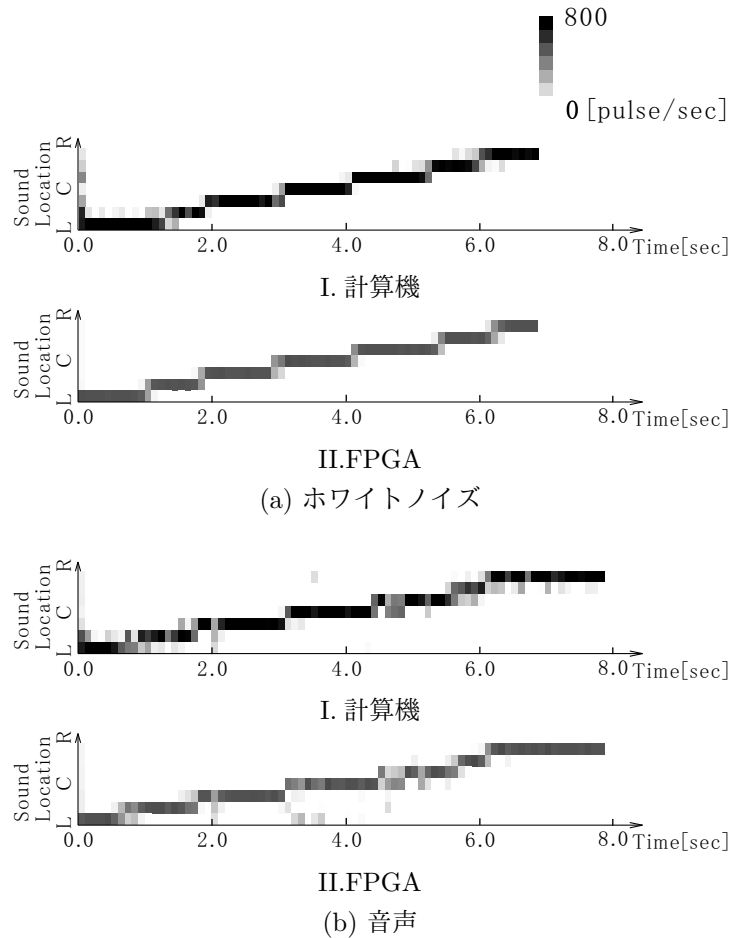


図 5.6: 音源位置識別部における出力結果

り下から音源位置の左→中央→右と対応している．各音とも音源の移動にあわせて対応した素子が多く発火しており，音源の定位ができているといえる．FPGAにおける結果は計算機シミュレーションにおける結果と比べて発火のばらつきが少ない．この理由はFPGAの演算では固定小数点による演算を行っているため，浮動小数点演算である計算機シミュレーションより計算精度が細くないためであると考えられる．しかし定位精度に大きな影響はないため今回実装したFPGAによる演算でも十分利用可能であるといえる．

今回の実験では学習データにはホワイトノイズしか用いていないが音声などの他の音に関しても定位できている．この理由は，CONPへ入力されるパターンは各周波数帯毎の時間差情報であり，音が違っていても出力される周波数帯が違っただけで時間差が同じであれば同じ箇所に入力される．よって全ての周波数成分を含むホワイトノイズの時間差パターンで学習を行えば部分的な周波数成分を持つ音の時間差パターンを包括することになり，各

表 5.5: 音源種類識別部の出力結果

Input Sound	Output Rate (%)					
	alarm bell	interphone	kettle	phone	voice	white noise
alarm bell	<u>99.5</u>	0.0	0.0	0.4	0.0	0.1
interphone	0.0	<u>96.9</u>	0.0	0.3	1.5	1.3
kettle	0.0	0.0	<u>99.5</u>	0.5	0.0	0.0
phone	0.0	0.0	0.0	<u>99.0</u>	0.0	1.0
voice	0.0	0.7	0.0	0.5	<u>98.8</u>	0.0
white noise	0.2	0.2	0.2	0.2	0.2	<u>99.0</u>

方向を学習した重みが最も近くなることが保証されるからである。

5.5 音源種類識別実験

音源種類の識別を行うにはまず周波数-パルス変換部で得られたパルス出力を周波数パターン量子化部へ入力しベクトル量子化を行う。次に得られたベクトル量子化されたパターンを音源種類識別部に与えて音源種類の識別を行う。音源定位実験と同じ音を入力した場合の FPGA 上での音源種類識別部の各音に対応するニューロンの出力頻度の割合を表 5.5 に示す。

表 5.5 よりそれぞれ各入力音にあわせて対応した素子が最も多く出力しており正しく識別ができていることがわかる。理想的には計算機シミュレーションにおける結果のように対応した素子のみが多く出力することが望ましいが、FPGA における結果では雑踏などの素子も多く発火している。これは FPGA 上では膜電位や内部電位を固定小数点で表現しているため、各段における CONP において計算誤差が多くなっていたことが原因である。しかし最大頻度となる出力素子は入力音と対応しており、システムの最終的な出力として最大頻度となる出力素子のみを考慮すれば実用上は問題ない。

5.6 システムの耐雑音性

実際の環境で使用することを考えた時、システムのノイズに対する耐性は非常に重要な点の一つである。そこで、録音した音に対して SN 比が一定値になるような大きさのノイズを左右両方に重畳し先の実験と同じ条件で定位及び識別を行わせた。

ノイズを重畳した場合の左 60 度における定位結果を図 5.7 に示す。図の横軸は SN 比を示し、縦軸は出力頻度の割合を示す。SN 比が 5dB になると

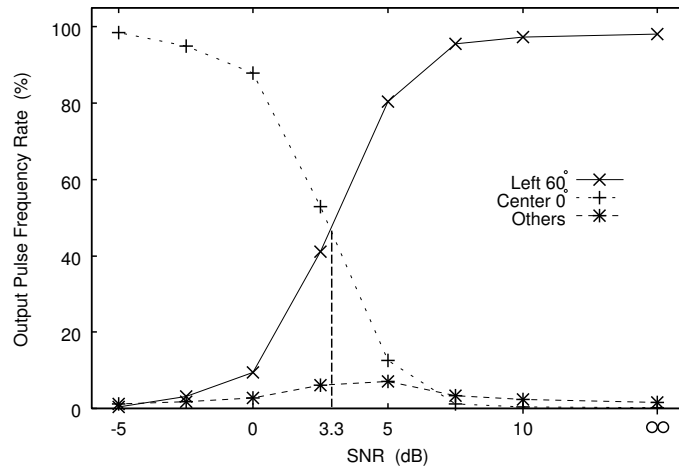


図 5.7: 音源定位部の耐雑音性

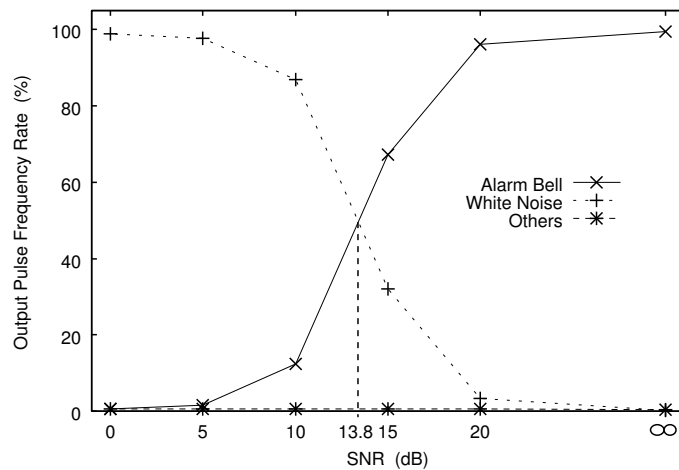


図 5.8: 音源種類識別部の耐雑音性

正しい位置を示す識別率が下がり、中央を示すようになる。この原因は入力パルス信号に後から含めた雑音の量が増加することで時間差検出部での左右のパルス信号の衝突判定が中央よりになり正確な時間差が検出できなくなるからである。

次に音源識別の結果を図 5.8 に示す。識別においても SN 比が小さくなるにつれて割合がノイズの方へと変化していく。これは雑音が大きくなるにつれて入力の周波数パターンが雑音に近くなり、量子化部における出力が雑音を示すようになるからである。

音源定位及び音源種類同定の両方の結果において SN 比が減少するに従い識別率は悪くなる。これはノイズが重畳するに従い時間差及び周波数スペクトルがノイズの特性になっていくからである。結果より音源定位におい

表 5.6: 各部の処理時間の比較

部位	計算機	
	シミュレーション [s]	FPGA [s]
入力音長	8.76	8.76
時間差検出部	36.82	0.83
音源位置識別部	4.59	0.85
周波数パターン量子化部	1.11	0.71
音源種類識別部	0.36	0.61
全動作	42.28	1.09

表 5.7: ハードウェア実装時に並列化が可能な演算個数

部位	個数
時間差検出部	14,480
音源位置識別部	2,534
周波数パターン分類部	493
音源種類識別部	82

てはSN比が3.3dB, 音源種類識別においてはSN比が13.8dBまで定位や種類の識別を正確に行うことができることがわかる. さらに精度を向上させるには背景雑音から目的音を抽出する方法などをシステムに適用することがあげられる. その方法の一つとして各周波数毎の時間差情報から目的音を抽出して定位と同定を行う方法を次章にて説明する.

5.7 処理時間及び回路数の比較

PNモデルは演算が単純であり各素子が独立・非同期で動作することができるためハードウェア上に実装することで非常に高速な演算を期待できる. これを確認するため入力音として音長が8.76[s]のホワイトノイズを用い本システムを動作させた場合の, 計算機シミュレーションとFPGAにおける各部の単独の処理時間及び全動作を行った場合の処理時間を表5.6に示す. なお計算機シミュレーションで利用した環境は表5.1の通りである. また表5.7にハードウェア上に実装した場合に並列に動作させることができる各PNモデルの個数や入力の個数などによる演算の個数を示す.

結果より, 音源種類識別部以外ではFPGAによる処理時間は計算機シミュレーションと比べて短い処理時間となっている. この原因は, 計算機上では表5.6に示した個数だけ計算を順次行わなければならないからである. 逆に音源種類識別部ではPNモデルの数が少ないため, FPGAに実装して並

列化するよりも高速な計算機で計算した方が処理が速くなる。しかし現実問題におけるパターン認識では多数の入力次元・識別数を必要とするため、表 5.6 中の「全動作」のように FPGA に実装することで計算機と FPGA の間に大きな処理時間の差を得ることが可能である。

続いて本研究で提案するシステムと CSP 法 [12][42], GMM による方法 [13][14] における実装上の回路規模について比較する。表 7.5 に各手法における単位時間あたりに並列化が可能な加算・積算の演算回数を示す。

ただし CSP 法に関しては高速化のために DFT の代わりに FFT を用い、 \cos などの演算については予め計算されたテーブルを用いる。時間窓の長さは $N=256$ とする。GMM 法に関しても Mel Frequency Cepstral Coefficients(MFCC) を計算する前処理部分に関して FFT とテーブルを用い、時間窓の長さは $N=256$ とする。MFCC の次元数は $L=26$ 次元とし、識別数は $C=6$ とする。提案システムに関しては計算機上で行われる周波数-パルス変換における演算も含み、PN モデルにおける減衰には積算ではなく 4.2 で説明したビットシフトと補数演算による実現とし 2 個分の加算と数えた。また使用する周波数帯域数を F , 時間差検出部の出力ニューロン数を M , 周波数パターン量子化部の出力ニューロン数を Q , 各識別数を C と表記する。鉤括弧内は本研究で FPGA に実装した部分を示す。

表 5.8 より各手法を個別に見た場合には提案システムの方が全回路の個数は多い。しかし提案システムでは CSP や GMM と比較して積算の個数が少ないことがわかる。ここで例として 16bit 同士の 2 つの数について加算と積算を行う回路を論理合成した場合の回路数について述べる。論理合成には本研究で実装に用いた開発ソフトウェアである Altera 社の QuartusII を使用した。この場合、加算を行う場合に必要な回路数が 16[ALUTs] になるのに対して、積算の場合には 235[ALUTs] の回路が必要となる。よって表 5.8 に示すように実装を行った場合、必要な回路数の概算は提案システムで 687,000[ALUTs] 程度、CSP 法と GMM 法を用いた場合で 2,076,000[ALUTs] 程度となり最終的な回路規模では提案システムの方が小さくなる。加えて提案システムにおいてはパルス変換までの前処理は同じ回路を用いることができるため、音源定位や音源種類の識別のために個別に手法を用いるよりも回路規模をさらに小さくすることが可能である。

以上、本節の実験結果から提案システムを FPGA に実装することにより、

表 5.8: 各手法における並列化した際に用いる演算回路の個数

提案システムの音源定位部分 (F = 15 , M = 21 , C = 7)	
加算	$2F \times 12 + 2F(10M+82)$
積算	$+ (C+1)(F \cdot M+4)+2(C+8) \rightarrow 11,702 \quad [11,312]$
	$2F \times 28 \rightarrow 840 \quad [0]$
提案システムの音源種類識別部分 (F = 43 , Q = 10 , C = 6)	
加算	$F \times 12$
	$+ (Q+1)(F+4)+2(Q+8)$
積算	$+ (C+1)(Q+4)+2(C+8) \rightarrow 1,195 \quad [679]$
	$F \times 28 \rightarrow 1,204 \quad [0]$
CSP 法 (N = 256)	
加算	$2 + 2N + 4 \rightarrow 518$
積算	$2N \log_2 N + 7N + 2N \rightarrow 6,400$
GMM 法 (N = 256 , C = 6 , L = 26)	
加算	$L+1+C(L+1)+C+1 \rightarrow 196$
積算	$N \log_2 N + 4L$
	$+ 2C^2 + C(L+1) + 1 \rightarrow 2,387$

計算機シミュレーションと比較して非常に短い時間で結果を出力することができ、入力信号に対してリアルタイムに結果を出力することも可能であり、また、従来手法と比較して少ない回路数で実装できることが確認できた。

5.8 まとめ

本章ではパルスニューロンモデルを FPGA 上に実装して時間差検出モデル、競合学習ネットワークを構成し、入力される音情報から音源定位と音源種類の識別を同時に行うことのできる聴覚情報処理システムを構築した。実験の結果、実装したシステムが計算機シミュレーションと同様に音源定位と音源種類の識別ができることを示し、さらにパルスニューロンモデルをハードウェア化することにより処理時間が大幅に短縮されリアルタイムでの動作が可能であり、さらに従来手法と比較して少ない回路数で実装できることが明らかになった。

本システムは、基本素子であるパルスニューロンモデルの組み合わせのみで構成されており、組み合わせの方法や学習の内容によって音以外の識別を行うことも可能である。さらにハードウェア化により高速な処理が可能であ

るため様々な機器への応用が期待できる。

聴覚障害者支援システムとして実際の環境で動作させることを想定すると複数の音源が存在する場合への対処が必要である。これについては、各音の発生する時間さえ異なれば認識部分に黒柳らの提案した方法 [36] を適用することで対応できると考えられ、今後の検討課題としたい。

第6章

複数音源分離及び音源種類識別 実験

前章では音源定位と音源種類の識別を行う聴覚情報処理システムをFPGAを用いて実装し実際に音情報からの実験を行い目的音の定位と種類の同定を高精度にできることを確認した。しかし実環境においては背景雑音や複数の音源など目的とする音以外の音源が存在するため識別精度が低下することが予測される。このような問題に対処するため複数の音源を分離することで目的音を個別に識別や定位を行う手法が提案されている [42][43]。しかしこれらの方法ではマイクロホンアレイを用いていることや計算方法が異なるためにPNモデルに直接導入することが難しい。そこで本章では、PNモデルを用いた複数音源を分離し、分離した音源毎に音源種類の識別を行う方法について説明する。音源の分離には音源から二つのマイクに与えられる時間差を利用する。

6.1 システム構成

複数音源分離及び音源種類識別を行うためのシステムの構成を図 6.1 に示す。周波数-パルス変換部、時間差検出部、音源種類識別部からなり、各部の構成は前章での音源定位及び音源種類識別システムと同じである。ただし時間差検出部の全ての出力を識別部の入力とするのではなく、同じ時間差を示す出力のみを識別部の入力とする点が異なる。また提案システムの動作例を図 6.2 に示す。それぞれ別の方向に存在する音源は混在した形でシステムに入力される。提案システムは周波数-パルス変換を行った後に時間差検出を行い、その時間差情報から音源の種類識別を行う。

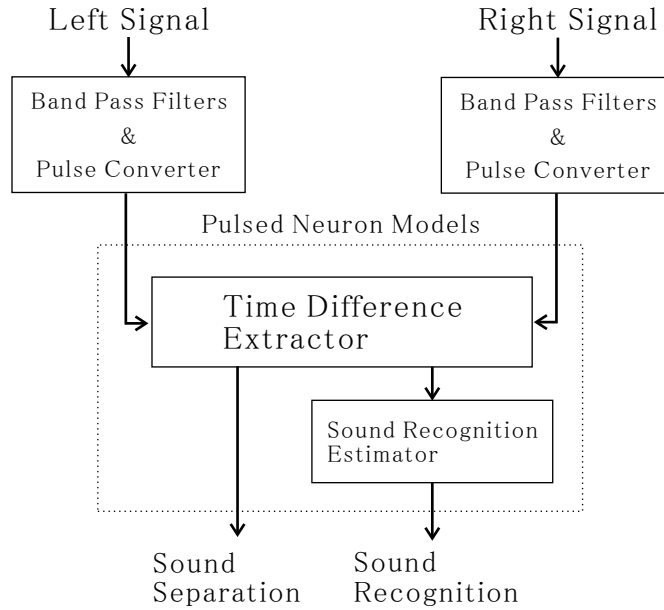


図 6.1: 複数音源分離及び音源種類識別を行うシステムの概略

6.2 実験条件

この実験では全て計算機上で作成した音を使用する。音の種類としては、単一の周波数の音として 500Hz, 1kHz, 2kHz の信号を用意した。また楽器の音として “Accordion”, “Flute”, “Piano”, “Drum”, “Violin” の 5 種類を用意した。また各音に対して 3 種類の時間差 “-0.5 ms”, “0.0 ms”, “+0.5 ms” を与えた。この時間差は図 6.3 のようにマイク間距離を人間の左右の耳の距離を想定して 17cm とし、音源からの距離を 1m として設定されたものである。それぞれの時間差 “-0.5 ms”, “0.0 ms”, “+0.5 ms” は音源位置が “左 90°”, “中央 0°”, “右 90°” に対応することになる。

6.3 時間差検出による音源分離

初めに音源を分離するための時間差について検出する。実験における各種パラメータは表 6.1 の通りである。サンプリング周波数を 48kHz としたため単位時間 Δt は $\Delta t = 1/48k = 20.8\mu[s]$ となり時間差検出部における出力ニューロンは 1 個あたり $41.6\mu s$ の時間差を示すことになる。

入力に単一周波数信号として +0.5ms の時間差を与えた 500Hz の信号, 0.0ms の時間差を与えた 1kHz の信号 -0.5ms の時間差を与えた 2kHz の信号の 3 種類を与えた場合の時間差検出部における出力を図 6.4(a) に示す。図の横軸は時間差検出部の出力ニューロン番号から計算した時間差を表し、縦軸

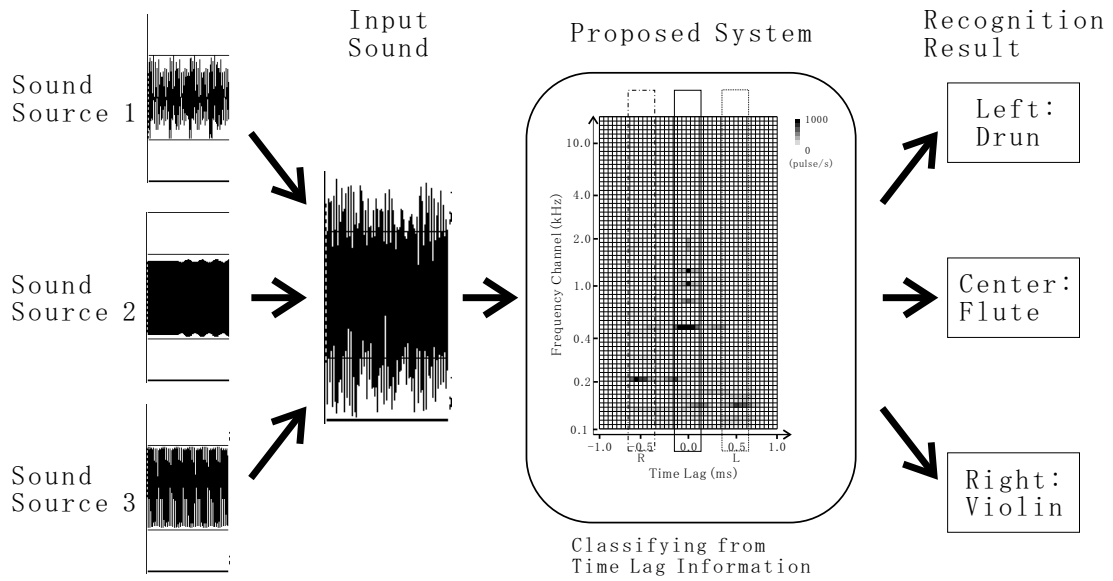


図 6.2: 提案システムの動作例

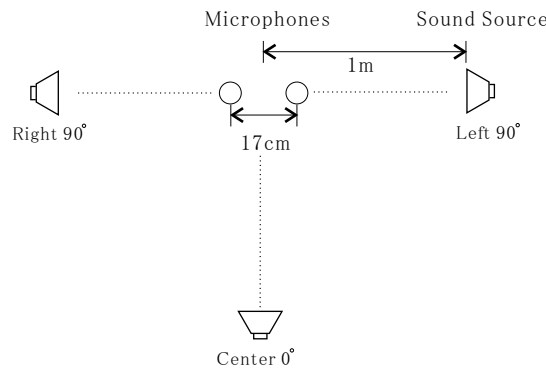


図 6.3: 実験における音源位置の状況

は周波数帯域を表す。また各ますの黒白の濃淡は各出力ニューロンのパルス出力頻度を示し黒いほど多くのパルスを出力している。結果より、大きく分けて $+0.5\text{ms}$, 0.0ms , -0.5ms の3種類の時間差を示すニューロンにおいて高い頻度で出力していることがわかる。また各時間差における発火頻度は入力音の周波数成分に従っている。同様に楽器の音として $+0.5\text{ms}$ の時間差を与えた“Drum”の音、 0.0ms の時間差を与えた“Flute”の音、 -0.5ms の時間差を与えた“Violin”の音の3種類を与えた場合の時間差検出部における出力を図6.4(b)に示す。この場合も大きく分けて $+0.5\text{ms}$, 0.0ms , -0.5ms の3種類の時間差を示すニューロンにおいて高い頻度で出力していることがわかる。

ここで図6.4(b)に示す時間差検出結果において、図6.5(a)のように $+0.5\text{ms}$,

表 6.1: 音圧差検出実験における各パラメータ

入力音	
サンプリング周波数	48 [kHz]
量子化ビット数	16 [bit]
周波数チャンネル数	72 [channel]
時間差検出部	
総シフト数	121
出力ニューロン数	41
MSO ニューロン閾値 θ_{TDE}	1.0
時定数 τ_{TDE}	350 [μ s]

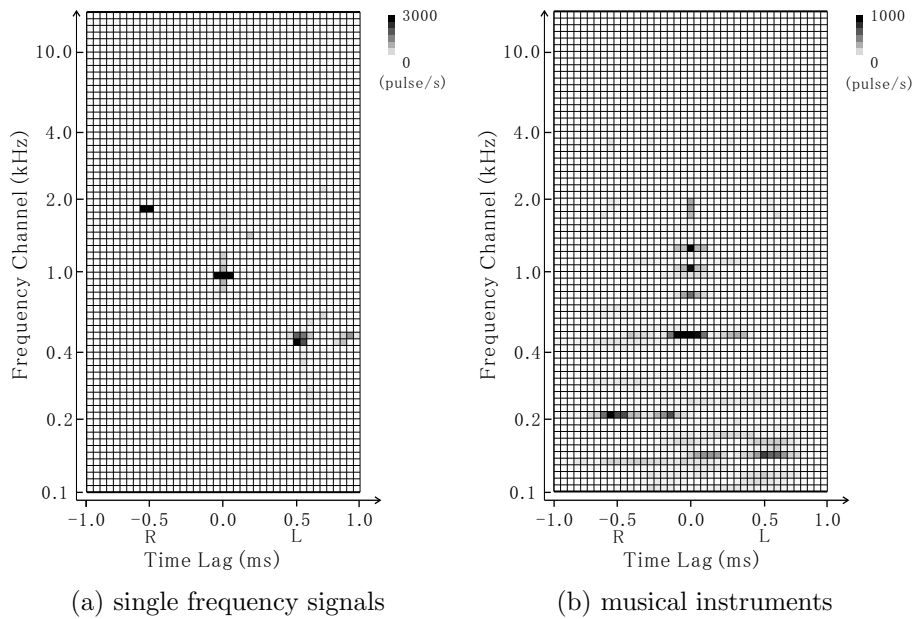
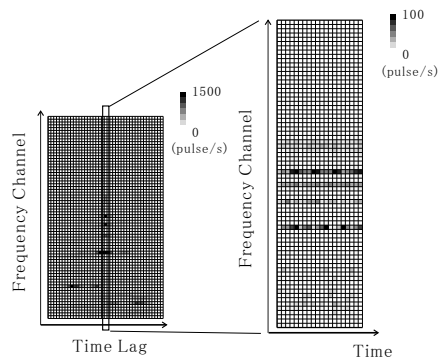
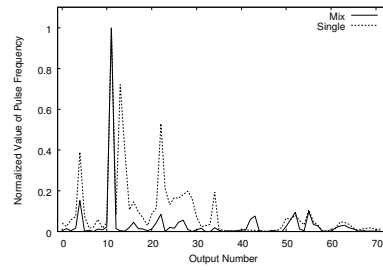


図 6.4: 複数の音を与えた場合の時間差検出部の出力結果

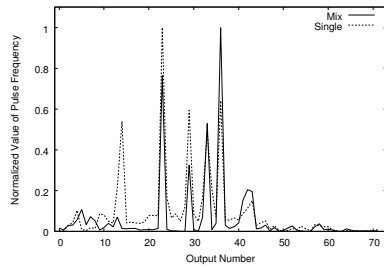
0.0ms, -0.5 msの時間差における各周波数毎の発火頻度を参照する. 図6.5(b)から(d)はそれぞれの時間差における発火頻度を示す. また時間差検出部の出力から示したもの(図中 Mix)の他に, 各楽器音について時間差を与えずに時間差検出部に入力した場合の0.0msにおける出力の結果(図中 Single)も示す. なお, 両者とも最大値が1.0, 最小値が0.0を示すように正規化されている. この結果より, 各時間差における出力の周波数パターンは基となる時間差を与えていない場合の出力の周波数パターンに非常に類似していることがわかる. よって各時間差の周波数パターンを調べれば, その時間差をもつ音が何の音であるかを識別できると推測できる.



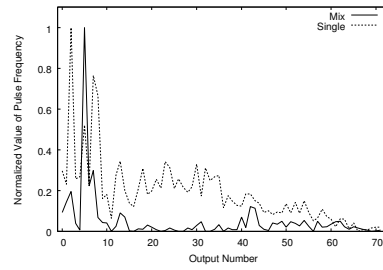
(a) Extraction of a time lag firing information



(b) Time Lag = -0.5 ms (Violin)



(c) Time Lag = 0.0 ms (Flute)



(d) Time Lag = +0.5 ms (Drum)

図 6.5: 各時間差における出力の周波数パターンの抽出

表 6.2: 複数音源分離実験における CONP のパラメータ

競合学習ニューロン	
入力数	72
競合学習ニューロン数	5[units]
閾値 θ	1.0×10^{-4}
入力閾値 θ_{gate}	100.0
入力パルス頻度からの減算率 β	0.11785
時定数 τ_p	20[ms]
不応期 t_{ndti}	10[ms]
無発火検出ニューロン / 複数発火検出ニューロン	
時定数 τ_{no} / τ_{multi}	0.5 / 1.0[ms]
閾値 $\theta_{no} / \theta_{multi}$	$-1.0 \times 10^{-3} / 2.0$
CL ニューロンへの結合重み	0.8 / -0.8
競合学習パラメータ	
学習係数 α	2.0×10^{-7}
学習回数	1000
学習半径の収束回数	400

6.4 分離した各音源の音源種類識別実験

前節の実験より、各時間差毎の出力から音源の種類が識別ができることが推測できるため、実際に CONP を用いて音源の識別を行う。初めに、時間差を与えていない 5 種類の楽器の音について時間差検出を行い、0.0ms における各周波数毎の出力パターンを用意する。この出力パターンを学習音として用い表 6.2 のパラメータに従い音源種類識別部の CONP の学習を行った。

学習後、識別実験として 5 種類の音それぞれに 3 種類の時間差を与え、合計 60 通りの組み合わせについて音源種類識別部に入力し識別を行った。ただし同時に存在する音は 3 方向でありそれぞれ異なる楽器であるとする。一例として、入力音に +0.5ms の時間差を与えた “Drum” の音、0.0ms の時間差を与えた “Flute” の音、-0.5ms の時間差を与えた “Violin” の音の 3 種類を与えて時間差検出を行い、各時間差における周波数パターンを音源種類識別部に入力した場合の各音を表すニューロンの発火頻度を図 6.6 に示す。時間差検出部の出力頻度に差があるため音源種類識別部における出力頻度にも大きな差があるが、各時間差毎に最も強く発火しているニューロンは適切な音源を示している。また -0.8ms などの入力に与えていない時間差においては音源種類識別部の出力もほぼ 0 となっている。ここから、入力信号に与えた時間差が既知でなくても時間差検出部の出力を個別に音源種類

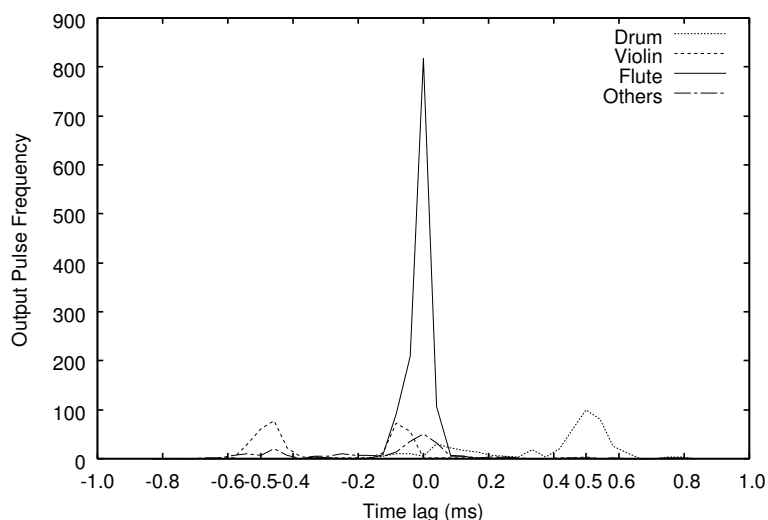


図 6.6: 各時間差における識別部の出力頻度

表 6.3: 音源種類の識別結果

Input \ Time Lag	Recognition Rate[%]		
	-0.5ms	0.0ms	+0.5ms
Acordion	89.9	88.1	88.8
Flute	92.3	94.4	92.4
Piano	62.5	32.9	64.0
Drum	90.3	89.1	88.6
Violin	79.7	78.4	79.0

識別部に入力し、全ての出力を参照することで音源の種類とその音源に与えられている時間差を判断することができる。

また表 6.3 は各音が各時間差を与えられた場合の識別率の平均を示す。ただし各識別率は各音を入力した際に発火した全発火数のうち適したニューロンが発火した数のことと定義する。

結果より“Piano”が0.0msの時間差を与えられた場合の結果が低い識別率となっている。この理由の説明として学習後のCONPにおける“Accordion”, “Flute”, “Piano”を示すニューロンの重み値を図 6.7 に示す。図より“Piano”を示すニューロンの重みのうち横軸4番や23番のように大きな値を持つ部分が他の楽器を示すニューロンでも大きな値になっていることがわかる。すなわち入力音の周波数成分に重なりがあるために他の楽器として識別されることが多くなる。また“Piano”の音は他の音と異なり、断続的な音であるため音の存在する短い時間で学習を行う必要があることも原因として考えられる。しかし“Piano”が0.0msの時間差を与えられた場合の結果以外に

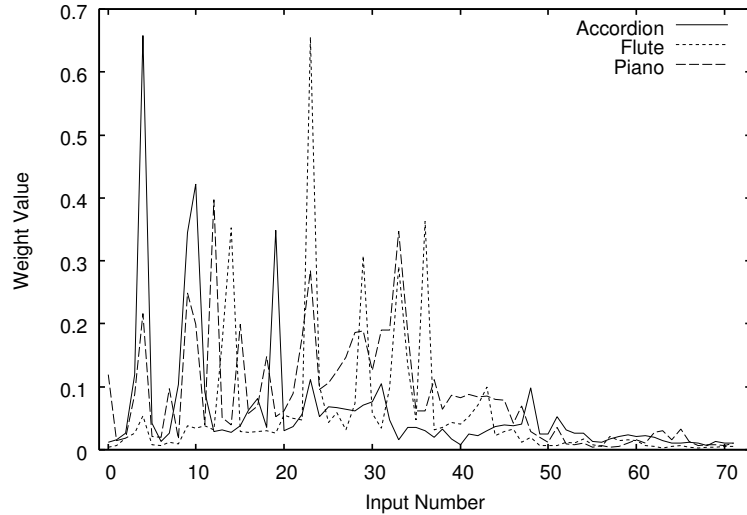
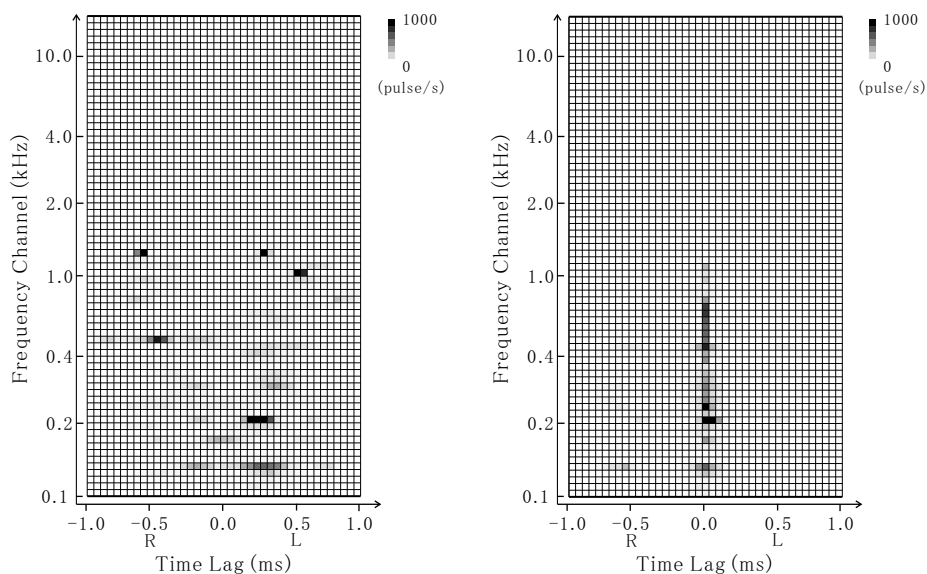


図 6.7: 学習後の各ニューロンの重み値

については78%を越える高い識別率が得られた。すなわち提案システムが時間差情報を基にして複数音源の識別を行うことができるといえる。

次に3方向のうち2方向の音を同じ楽器にして実験を行った。入力音として+0.5msの時間差を与えた“Flute”の音、0.0msの時間差を与えた“Violin”の音、-0.5msの時間差を与えた“Violin”の音の3種類の音を与えた場合の時間差検出部の出力結果を図6.8(a)に示す。また、比較のために0.0msの時間差を与えた“Violin”の音だけ入力に与えた場合の時間差検出部の出力結果を図6.8(b)に示す。図6.8(a)より入力音として3種類の時間差を与えたにも関わらず、時間差検出部においてパルス頻度の大きな時間差は+0.5msと-0.25msの2つしか検出できていない。-0.25msについては、周波数パターンが図6.8(b)に類似しているため-0.25msに表れている音は“Violin”であると判断できる。すなわち0.0msの時間差を与えた“Violin”の音と-0.5msの時間差を与えた“Violin”の音が-0.25msの時間差をもつ一つの音として検出されたことになる。実際に人間は同じ音を別の方向から同時に与えられた場合、脳内ではその中央に定位されておりこれと同じ挙動を示したといえ、本システムは聴覚情報処理を基にしているため妥当な結果であるといえる。



(a) two identical “Violin” signals on the left and central positions and “Flute” signal on the right position

(b) single “Violin” signal on the central position

図 6.8: 二方向に同じ音を用いた場合の時間差検出部の出力結果

6.5 まとめ

本章では本論文で提案する聴覚情報処理システムにおいて、識別率の低下の原因となる複数の音源が入力された場合に各音源を分離し個別に音源の種類を識別するための方法について説明した。音源の分離には音源から二つのマイクに与えられる時間差を利用する。また時間差検出部の出力を周波数パターンとして見た場合基の音源の特性に沿ったものになるため各時間差における音源の種類を識別することができる。以上の方法を用いることで提案手法が各音源を分離し個別に音源の種類を識別することができることを実験により確認した。

本手法では音源を分離するための条件として、入力音の空間的位置が異なり二つのマイクの間時間に時間差が生じる必要がある。また入力音の周波数成分に重なりがある場合には識別が難しいという問題が残る。これに対して、黒柳らによって音の発生する時間に差がある場合にPNモデルを用いて各音を分離する方法が提案されている [36]。他に木下らによる周波数成分の重なりに適応する処理 [44] を加えるなどの方法を用い、より精度を上げていく必要がある。

また本研究で提案する手法は適用するための特殊な素子は必要なく実現することが非常に容易であるといえる。ただし実験のように、時間差検出部

の各時間差における出力パターン全てを音源種類識別部に並列に入力することは回路規模の点から非現実的であるため、出力の頻度から識別を行う時間差部分を選択するための機構を用意するなどの工夫は必要となる。このような機構などを含め、実際にFPGA上に実装し他手法との精度などを比較することが今後の課題である。

第7章

音源接近検出及び音源種類識別 実験

本章では、前章までと同じFPGAに実装したPNモデルを用い、その組み合わせを変更するだけで他の聴覚情報処理を実現できることを示すために、音圧差検出モデルと競合学習ネットワークによる音源接近検出及び音源種類識別実験について説明する。本論文の中ではこの音源接近検出及び音源種類識別システムの目的として車両の接近検出及び種類の識別を行うこととし、各種実験を行った。

音源の接近を検出する手法としては、黒柳らの提案した音圧差検出手法[30]を用いる。黒柳らの手法では音源定位を行うために左右二つの信号の間の音圧差を検出しているが、本システムでは、入力される一つの音信号のうちある周波数成分 f に関するパルス列 $x_f(t)$ と、同じパルス列に一定の遅れ時間 T_D を与えたパルス列 $x_f(t-T_D)$ を音圧差検出部に与えて音圧差情報を検出する。検出の結果、前の時間における音圧より現在の時間における音圧の方が大きい場合、対象となる音源が接近していると判断する。

7.1 システム構成

本章で用いる音源接近検出及び音源種類識別システムについて概略図を図7.1に示す。本システムは大きく分けて周波数-パルス変換部、音圧差検出部、音圧差情報量子化部、音源種類及び接近識別部からなる。音圧差検出部、音圧差情報量子化部、音源種類及び接近識別部についてはPNモデルで構成されており、FPGA上に実装されている。なお本論文においては周波数-パルス変換部及び時間遅れ T_D を与える部分については計算機で処理される。

入力された音信号は計算機上の周波数-パルス変換部で複数の周波数成分に分割された後パルス情報へと変換される。変換されたパルス情報には並

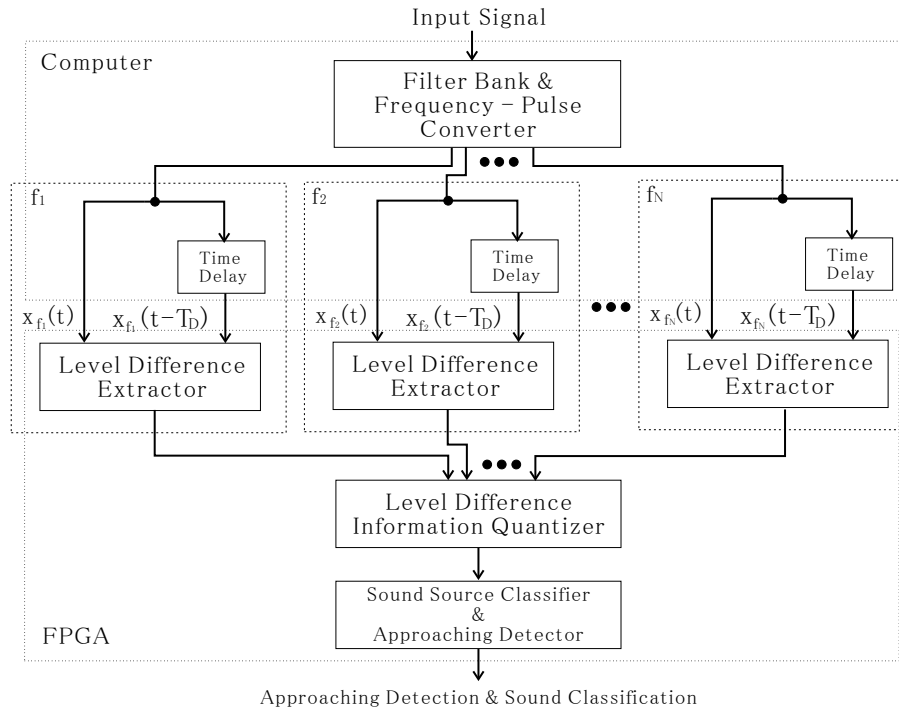


図 7.1: 音源接近検出及び音源種類識別を行うシステムの概略

行して時間遅れが与えられ、音圧差検出部に入力される。音圧差検出部で検出された音圧差情報を音圧差情報量子化部である競合学習ネットワークに入力することで量子化を行う。最後に量子化された情報から教師あり学習を行った競合学習ネットワークにて音源の種類を識別する。最後に得られた接近検出結果及び音源種類の識別結果は再び計算機へ返され、計算機上で出力結果を表示する。

7.2 実験条件

本実験では自車が交差点などで停止または徐行していた時に、死角から二輪車や緊急車両が接近している状況を想定する。このため学習用の音としては、図 7.2 のように自車が停止中に音源が時速 30km/h で 2 秒間で接近しその後 2 秒間で離脱すると仮定して音圧を変化させた音を計算機上で作成した。具体的には、最も接近した時に 1m 離れた所に位置しこの時の音圧を S_{base} として、各時間における音圧 $S(t)$ を距離 $d(t)$ の二乗に反比例させる。音源の種類としては“パトカーのサイレン音”，“救急車のサイレン音”，“スクーターのエンジン音”をの 3 種類と，“その他”として出力させるために“路上の背景雑音”を用意した。このため識別する数としては 3 種類の音の接

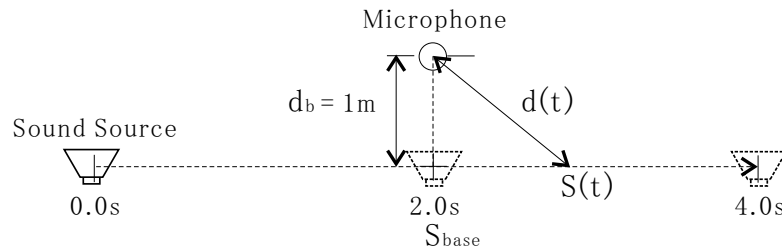


図 7.2: 学習実験に用いた音源の移動

近と離脱, その他の合計7種類となる. また本システムの実装に用いた計算機及びFPGAの仕様を表5.1に示す.

初めにこれらの音を用いて表7.1に示すパラメータを用いて計算機上で各CONPの学習を行った. このうちLSOニューロン, CLニューロンの時定数に関しては時間的な音圧差を検出する場合には0.1[s]程度の分解能があれば十分であると考えて設定し, 各ニューロン数に関してはFPGAに実装することが可能な最大の個数であることから表の値を設定した. その後, 学習の結果得られた重みや, 表7.1に示す競合学習パラメータ以外のパラメータの値に従いFPGA上に各部を実装した. 実装の際に必要なとした各部の回路数を表7.2にあげる. 合計が43,100[ALUTs]であるため使用した開発ボードに全ての回路を搭載することが可能である.

表 7.1: 音源接近検出及び音源種類識別実験におけるパラメータ

入力音	
サンプリング周波数	16[kHz]
量子化ビット数	8[bit]
周波数-パルス変換部	
用いる周波数帯域数	11[Channel]
音圧差検出部	
時間遅れ T_D	0.4[s]
LSO ニューロン総数 ($2N$)	42[個]
b	6
LSO ニューロン間隔	1[個]
LSO ニューロン重み係数 γ / ϵ	60 / 60
LM2 ニューロン総数 ($2L$)	20[個]
閾値 $\theta_{LSO} / \theta_{LM2}$	0.001 / 0.001
時定数 τ_{LSO} / τ_{LM2}	0.1[s] / 35[μ s]
不応期 RP_{LSO} / RP_{LM2}	1.0 / 1.0[ms]
競合学習ニューロン	
音圧差情報量子化部 CL ニューロン数	23[個]
音源種類及び接近識別部 CL ニューロン数	7[個]
閾値 θ	1.0×10^{-4}
制御閾値 θ_{gate}	50.0
入力パルス頻度からの減算率 β	3.125×10^{-2}
膜電位減衰の時定数 τ_p	0.1[s]
不応期 RP_{CL}	1.0[ms]
無発火検出ニューロン / 複数発火検出ニューロン	
時定数 τ_{no} / τ_{multi}	0.5 / 1.0[ms]
閾値 $\theta_{no} / \theta_{multi}$	$-1.0 \times 10^{-3} / 2.0$
CL ニューロンへの結合重み	20.0 / -20.0
競合学習パラメータ	
学習係数 α	6.0×10^{-7}
学習回数	1000
学習半径の収束回数	200

表 7.2: FPGA 上での各回路の回路数

部位	回路数 [ALUTs]
FPGA 入出力部	83
音圧差検出部	25,498
音圧差パターン量子化部	16,170
音源種類及び接近識別部	1,349
合計	43,100

7.3 音圧差検出実験

音源の接近検出を行うには音圧差検出部において現在の音信号と一定時間前の同じ音信号との音圧差を検出する。学習音の一つである“スクーターのエンジン音”に対する各時間間隔毎の音圧差検出部の出力を図7.3に示す。なお音圧差を検出する際に与える時間遅れ T_D は予備実験の結果から十分な音圧差が得られる最小時間として0.4[s]に設定してある。図の横軸は音圧差検出部の出力ニューロン番号 n 、縦軸に周波数帯域を示す。図の横軸の数値が大きいほど現時刻の入力 $x_f(t)$ が遅れ時間差を与えた入力 $x_f(t - T_D)$ より大きな音圧差であることを示す。また横軸のうち+側は $x_f(t)$ が $x_f(t - T_D)$ より大きいことを表し、音源が接近している (Approaching) ことを示す。逆に横軸の-側は $x_f(t)$ が $x_f(t - T_D)$ より小さいことを表し、音源が離脱している (Moving away) ことを示す。

結果より、音源の接近時(0.0s~2.0s)には横軸+側のニューロンが多く発火している。さらに音源がマイクにより近い1.0s~2.0sの時の方が横軸の数値が大きいニューロンが発火し、より接近していることを示している。逆に離脱時には-側のニューロンが多く発火しており、音源がマイクに近い2.0s~3.0sの時の方が数値が小さいニューロンが発火している。以上のことから、音源の移動に伴い音圧差検出部が正しく音圧差を検出できているといえる。

次に接近時及び離脱時における各学習音毎の音圧差検出部の出力を図7.4に示す。例えば“救急車のサイレン音”であれば約800Hzの周波数帯域において横軸の数値の大きなニューロンが多く発火しているように、それぞれの音に関して大きな音圧差を示す周波数帯域が異なることがわかる。そのため、音圧差検出部の出力パターンから音源の種類を推定することができると予測できる。

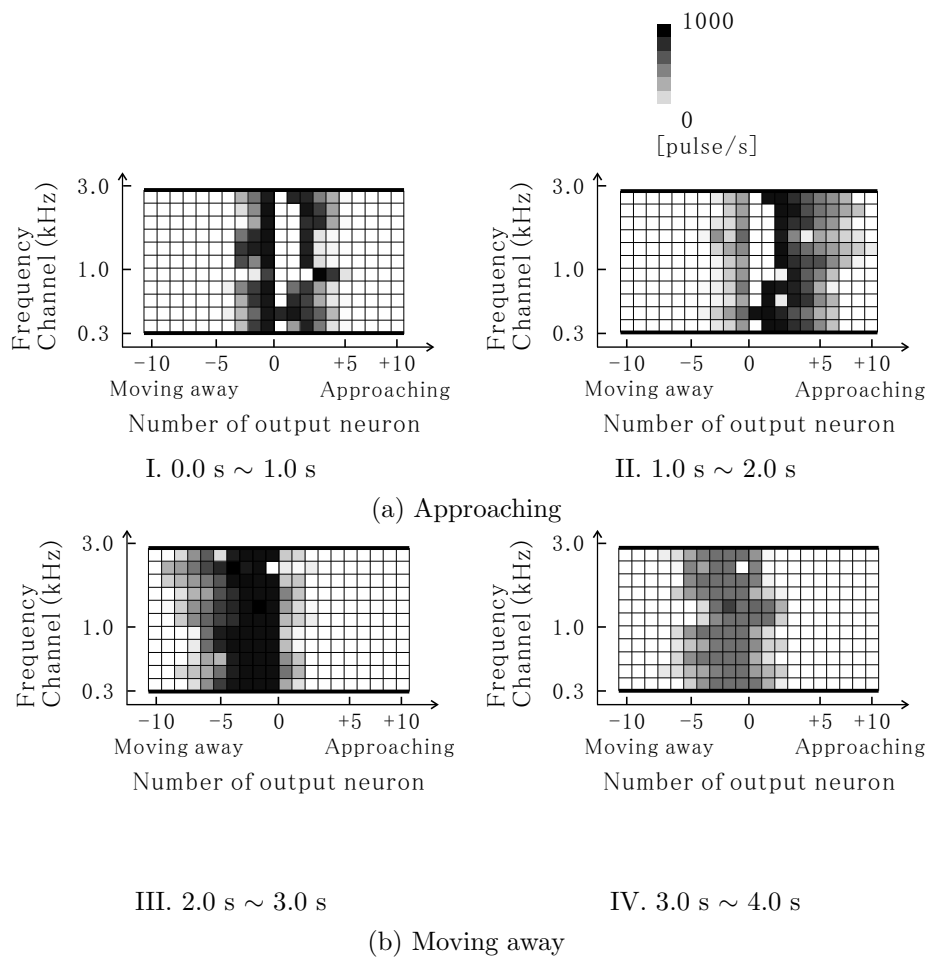


図 7.3: 各時間毎における音圧差検出部の出力結果

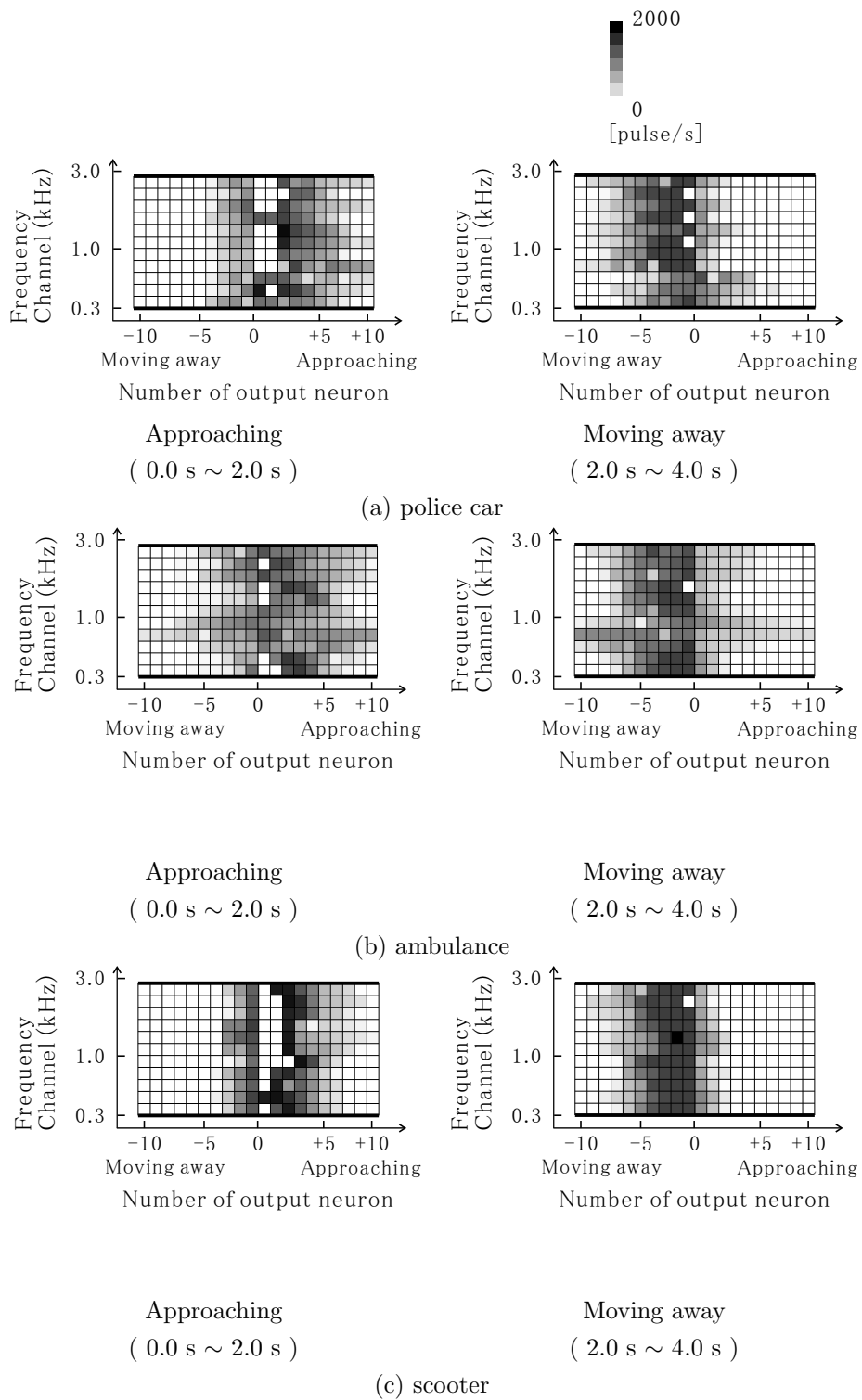


図 7.4: 各入力音毎における音圧差検出部の出力結果

表 7.3: 音源種類と接近識別部の認識結果

Input Sound		Recognition Rate[%]						
		police car		ambulance		scooter		other
		A	M	A	M	A	M	
police	A	<u>99.9</u>	0.0	0.0	0.1	0.0	0.0	0.0
	M	0.7	<u>99.3</u>	0.0	0.0	0.0	0.0	0.0
ambulance	A	3.2	0.0	<u>93.4</u>	0.0	3.2	0.0	0.0
	M	0.0	0.0	1.0	<u>99.0</u>	0.0	0.0	0.0
scooter	A	0.6	0.0	0.0	0.0	<u>99.6</u>	0.0	0.0
	M	0.0	0.0	0.0	0.0	1.8	<u>98.2</u>	0.0

7.4 音源種類及び接近識別実験

音源種類の識別を行うにはまず音圧差検出部の出力を音圧差情報量子化部へ入力しベクトル量子化を行う。次に量子化されたパターンを音源種類及び接近識別部に与えて音源の種類識別と接近の検出を行う。表 7.3 に学習音を入力した場合の FPGA 上における音源種類及び接近識別部の各ニューロンの出力頻度の割合を示す。下線は割合が最大となる出力結果を示す。

結果から学習用に用いた各入力音に対して適切なニューロンが最も高い頻度で出力しており、各音の接近・離脱を正しく学習し識別できているといえる。

7.5 実際の音に対する認識実験

次に本システムが実環境で利用できるかどうかを検証するために、実際に得られる音に関して同様に接近検出及び音源種類の識別実験を行った。ただし緊急車両に関しては音を収録することが困難であるため、本論文では自転車に対して二輪車が接近する状況のみに限定して実験を行った。

実験環境としては、晴天時のアスファルト上で行い、図 7.5 のように自転車の両側のルーフに SHURE 社製無指向性マイク “WL-93” を取り付ける。そして図 7.6 のようにアイドリング状態の自転車に対して左後方からスクーターが接近し通過するまでの音を録音する。

録音した音に関して提案手法を用いて識別実験を行った結果の一例を図 7.7(b) に示す。図の横軸は時間であり音源が真横にきた時間を 0.0[s] とする。縦軸は振幅と識別結果であり識別結果の濃淡は各ニューロンの出力パルス頻度を示し黒いほど多くのパルスを出力していることを表す。接近と離脱



図 7.5: 録音マイクの取付位置

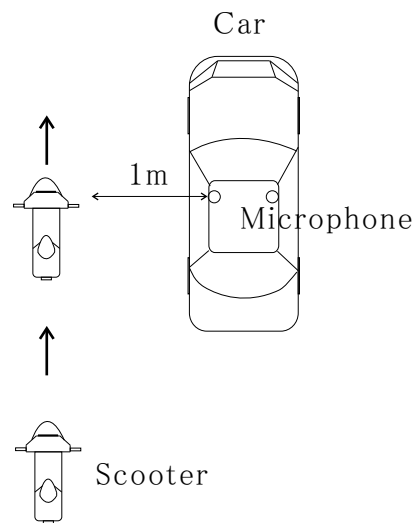


図 7.6: 路上音の録音状況

の識別が変わる時間が実際の音より遅れているが、これは音圧差検出部では一定時間前の入力と比較しているためである。

また学習データとして計算機上で仮想的に設定した音を用いたが、実験結果のように実環境で得られる音に関しても接近検出と識別ができています。これは識別部において勝者ニューロンのみが発火するように制御しているため、環境により入力音が変化しても、正答に対応する学習音と入力音との間の類似度が他の学習音よりも高ければ、正答に対応するニューロンのみが発火し正しく識別できるためと考えられる。このため提案手法は動作環境がある程度変化しても重みやパラメータを再設定することなく対応することができる。

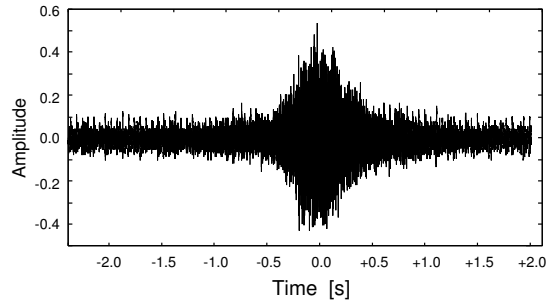
続いて比較実験として、まず接近検出については大塚らの手法[46]や星野の提案しているレベル閾値またはレベル上昇度から判断し検知する手法[47]のように音源の持つ特定の周波数信号のレベルが過去の時間より上昇する

ことから接近と判断する手法について同様の実験を行った。スクーターの特徴周波数として1.8kHzを選択しこの周波数帯域におけるレベルの時間変化を図7.7(c)に示す。さらに背景雑音と区別するために閾値を-45dBとし、閾値以上でありかつレベル値が過去の値より上昇している区間も図に示す。この区間では音源が接近していると判断する。結果では-1.6[s]より接近を検知できている。しかしスクーターが通りすぎた+1.2[s]から+1.4[s]においても接近と検出してしまうことになる。これは1.8kHzの周波数帯域のみのレベルの時間変化を参照しているためであると考えられる。

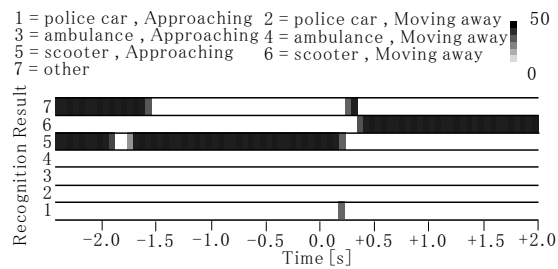
次に音源の検出と識別を行う手法として、星野の提案しているニューラルネットワークによる走行音の周波数スペクトルから検知する手法[47]についても実験を行った。実験条件としては文献[47]を参考に、本手法と同じく3種類の音源の識別も同時に行うために入力層のニューロン数を $I = 60$ 、中間層、出力層のニューロン数を3倍である $M = 15$ 、 $O = 3$ へと増やした3層の階層型ニューラルネットワークを用いると仮定し、文献[47]と同じく出力値を評価しやすくするために過去0.5[s]の間の出力を平均してスムージング処理を行い最終的な出力値とした。この時のスクーターを示すニューロンの出力値とその他のニューロンの出力値の合計を図7.7(d)に示す。結果より[47]の手法では出力値が0.5以上である-2.0[s]から“スクーター”として音源の接近を検出できる。ただし手法の都合上、出力値からは音源の接近・離脱及び停止を直接区別することはできず、これらを区別するにはレベル値による方法と併用したり出力値の時間的変化を見る必要がある。

これらの手法と比較して提案手法では、-1.5[s]より以前では“その他”を示すニューロンも交互に発火しており“スクーターの接近”として確実に識別できる時刻が従来手法より遅い。この理由は、提案手法では音源の接近検出及び種類の識別を同時に行うために周波数帯域毎の音圧の変化を用いているが、スクーターが遠方にある時には音の音圧変化が小さく、接近として識別ができないためである。しかし比較した従来手法においても接近及び離脱を検出するためには時間的な音圧の変化を計算する必要があり、この場合提案手法と同様に検出可能距離が短くなることが考えられる。また、人間が物体を認知してから対応するまでの時間は通常1.0[s]であるといわれており[45]、提案手法の実験結果でも1.5[s]前より接近の検出ができているため提案手法は検知システムとしては問題ないといえる。

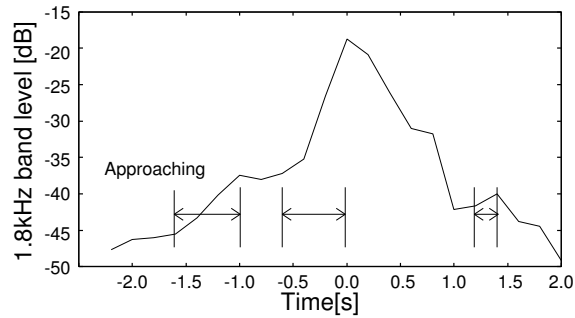
以上の比較実験から，提案手法は従来手法では個別に行っていた音源の接近及び離脱の検出と音源種類の識別を同時に行うことができ，また接近検知システムとして十分な識別精度をもつことが確認できた．



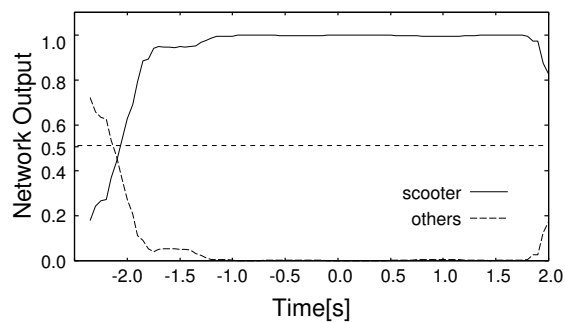
(a) 入力音



(b) 提案手法における出力結果



(c) レベル判断による手法の結果



(d) 周波数スペクトル識別による手法の結果

図 7.7: 路上音における出力結果

表 7.4: 各部の処理時間の比較

部位	提案手法		従来手法
	計算機 (s)	FPGA(s)	
入力音長	4.00	4.00	4.00
音圧差検出部	11.20	0.40	
音圧差情報量子化部	8.96	0.38	
音源種類及び接近識別部	0.42	0.37	
全動作	20.58	0.40	0.71

7.6 処理時間及び回路数の比較

交通安全装置の開発において重要な点として、出力を得るまでの応答時間がある。本システムにおける応答時間を検証するため、入力音として音長が4.00[s]のデータを用い本システムを動作させた場合の、FPGAにおける各部の単独の処理時間及び全動作を行った場合の処理時間を表7.4に示す。また比較のために、計算機シミュレーションにおける本手法の処理時間と、従来手法として星野の提案しているニューラルネットワークによる走行音の周波数スペクトルから接近を検知する手法[47]についても処理時間を示す。なお従来手法については先の実験と同じく入力層、中間層、出力層のニューロン数を $I = 60$, $M = 15$, $O = 3$ と仮定した3層の階層型ニューラルネットワークを用いた。また計算機シミュレーションで利用した環境は表5.1の通りである。ただし各手法ともにFFTによる周波数変換などの識別部に入力するデータを生成するまでの時間は考えない。

表7.4より、計算機を用いて処理を行った場合、提案手法は従来手法と比べて識別のための複数の手順があり順次計算する必要があるため計算時間が長くなる。しかし提案手法はFPGA上に実装することにより並列に計算することができるため大幅な速度の短縮ができていることがわかる。よってFPGA上に実装することにより提案手法でも従来手法と同じく入力音に対して十分短い時間で出力が得られ、リアルタイムに動作を行うことができる。

従来手法もハードウェア上に実装することで並列に計算が可能となり高速な計算が可能であると予測される。そこでそれぞれの手法における周波数変換などを含む全ての処理に必要な演算から、ハードウェア上にシステムとして実装した場合に必要な回路数の概算を表7.5に示す。従来手法ではFFTの窓長さ N を256とし計算に必要な係数は予め用意されているものと

表 7.5: 各手法における並列化した際に用いる演算回路の個数

	提案システム	従来システム
	周波数-パルス変換部	FFT + 正規化
積算	308	2,304
加算	132	257
	PN モデル部分	ニューラルネットワーク部分
積算	0	945
加算	7,598	36
必要回路数	196,060 [ALUTs]	768,203 [ALUTs]

する。このため積算の回数は正規化を含めて $N \log_2 N + N \rightarrow 2,304$ だけ必要とし、加算は $1 + N \rightarrow 257$ だけ必要となる。ニューラルネットワーク部分では各ニューロン数が入力層 $I = 60$ 、中間層 $M = 15$ 、出力層 $O = 3$ であるため積算が $I \cdot M + M \cdot O \rightarrow 955$ だけ行われ、閾値による比較も加算と同等の演算とすると総和と比較により加算の総数は $2M + 2O \rightarrow 36$ となる。これに対して提案手法では、使用する周波数帯域の数を $F = 11$ 、音圧差情報量子化部での量子化数を $Q = 23$ 、音源種類及び接近識別部での識別数を $C = 7$ とすると、周波数-パルス変換部で積算回数が $28F \rightarrow 308$ 、加算回数が $12F \rightarrow 132$ となる。続いて PN モデルにて実装された部分に関しては積算を必要とせず、加算が音圧差検出部で $174F \rightarrow 1,914$ 、音圧差情報量子化部で $(Q + 1)(20F + 4) + 2(Q + 8) \rightarrow 5,438$ 、音源種類及び接近識別部で $(C + 1)(Q + 4) + 2(C + 8) \rightarrow 246$ だけの加算が必要となる。

表 7.5 より提案システムの方が全回路の個数は多くなる。しかし提案システムでは従来システムと比較して積算の個数が少ないことがわかる。16bit 固定小数点同士の 2 つの数について加算と積算を行う回路を、本研究で実装に用いた開発ソフトウェアである Altera 社の QuartusII を使用して論理合成した場合、加算を行う場合に必要な回路数が 16[ALUTs] になるのに対して、積算の場合には 235[ALUTs] の回路が必要となる。よって表 7.5 に示すように実装を行った場合、必要な回路数の概算は提案システムは 196,000[ALUTs] 程度、従来システムは 768,000[ALUTs] 程度となり最終的な回路規模では提案システムの方が小さくできる。なお本実験では文献 [47] を参考に従来手法の中間層のニューロン数を 15 個と仮定したが、学習の状況によってはどちらの手法においてもさらに多くのニューロン数を必要とすることがあり得る。この場合従来手法では積算の回数が増えるため、提案手法と従来手

法の回路数の差はさらに広がり、提案手法の方が有利になる。

以上の実験から、提案システムは従来システムでは個別に行っていた音源の接近と種類の識別を同時に行うことが可能であり、かつ従来システムより少ない回路数でハードウェアに実装でき、入力信号に対してリアルタイムに結果を出力できることが確認できた。

7.7 まとめ

本章では、自転車に対して接近する二輪車や緊急車両を音を用いて検知する方法として、FPGA 上に実装されたパルスニューロンモデルを用いて新たに音圧差検出モデル、競合学習ネットワークを構成し、入力される音情報から周波数毎に音源の時間的な音圧差の情報を抽出することで音源の接近の検出と接近する音源の種類を識別するシステムを提案した。実験の結果、提案システムが従来システムでは個別に行っていた音源の接近・離脱の検出と接近する音源種類の識別を同時に行うことができ、かつ従来システムより少ない回路数で実現できることを示した。

音による車両の接近検知手法は障害物の多い街路地や交差点、夜間などの環境において有利であるが周囲の雑音が非常に大きい場合などには不利である。結果としてカメラからの画像処理による手法 [48][49] やセンサを用いた手法 [50] のような、視覚的手段と音を用いた聴覚的手段を併用することが最適であると考えられる。視覚的・聴覚的手段を併用する場合にはそれぞれのシステムの装置規模ができるだけ小さく安価であることが望ましい。実験結果で示したように、本システムは従来手法より回路規模が小さく併用する際にも有効な方法であるといえる。また基本素子である PN モデルの構成は先に提案した音源定位と音源種類の識別を行う回路 [37] と同じであり、識別手法毎に個別の回路を製作する必要がなく一つの装置にまとめることが容易であるため、最終的な装置の規模を小さくできる。加えて併用時には各識別結果を相互に反映させることが容易であり、性能の向上も期待できる。

今後の課題としては、実際の環境では接近する車両が複数ある場合など音環境が大きく変化するため、実際に車載システムとして搭載して検証を行い対策を検討することがあげられる。

第8章

あとがき

本研究では、音情報を利用した聴覚障害者への支援装置などの開発を目的に、パルスニューロンモデル(以下PNモデル)を用いて音源位置の定位や音源種類の同定などを行う聴覚情報処理システムについて、FPGAを用いて実装を行いその動作について検証した。

1章では本研究の背景について述べると共に問題に対する本研究の目的と手法の概要を記述した。また本論文の構成について説明した。

2章では本研究で提案する聴覚情報処理システムの基となる人間の聴覚機構について構造と機能を説明した。

3章では聴覚情報処理システムにおける各部位の動作を説明した。まず与えられた入力は、フィルタ群によって各周波数成分に分割される。次に各周波数成分毎に非線形変換が行われた後、振幅値にそった値でパルス情報に変換される。このパルスに変換された各周波数毎の情報をPNモデルで構成された検出部や識別部に入力することで各種聴覚情報処理を行う。PNモデルは時間的な情報を扱うのに適しておりハードウェア化が容易である。

4章では聴覚情報処理システムをFPGAで実装するための方法について説明した。通常のニューロンモデルにおいては入力と重みの積を計算する必要があるが、PNモデルは入出力が0か1のパルス列であるためこれを入力パルスと重みの論理積で実現できる。通常のニューロンモデルにはない時定数による電位の減衰に関しては、ビットシフトと補数演算を利用することで実現する。そして時間差検出部や音圧差検出部、CONPの構成について説明した。

5章では、実際にハードウェアに実装したPNモデルを組み合わせ、音源定位と音源種類識別を行う聴覚情報処理システムをFPGA上に構築し、検証を行った。音源定位を行うには、入力パルスの時間差をPNモデルで構成

された時間差検出モデルにて検出し、PNモデルで構成された競合学習ネットワークにて位置を定位する。音源種類の識別を行うには周波数パターンを用い、競合学習ネットワークにて識別を行う。実験の結果、FPGA上に実装した本システムが音源定位と音源種類の識別を行うことができることが確認できた。ハードウェアに実装することにより、計算速度が大幅に向上し入力音に対してリアルタイムに結果を出力することができるようになった。またPNモデルによる本手法は他手法と比較して小さな回路規模でハードウェアに実装できることが確認できた。

6章では背景雑音や複数の音源がある環境に対処するために聴覚情報処理システムを用いた複数音源の分離と定位、また個別に識別を行う方法について説明した。音源を分離するための方法としては入力音の時間差を利用する。計算機上でのシミュレーションの結果、時間差検出部の出力は各音の持つ周波数情報に依存するため各音の時間差及び種類の識別を行うことができることが確認できた。

7章ではPNモデルの組み合わせを変更し別の聴覚情報処理の実現として音源の接近検出と種類の識別を行うシステムを構成した。音源の接近検出には時間の経過による音圧の変化を利用し、PNモデルで構成された音圧差検出モデルにて過去の時間と現在の時間の間の音圧差を検出する。また音圧差の検出を各周波数毎に行うことによって音源の持つ特徴周波数の音圧差変化を調べ、その周波数パターンから音源の種類を識別する。実験では提案システムの利用先として屋外での車両接近検出を想定し、自動二輪車の接近検出について検証した。実験の結果、提案システムが音源の接近検出と種類の識別を行うことができ、他手法よりも小さな回路規模で実現できることが確認できた。

以上の実験から、FPGA上に実装したPNモデルを組み合わせることで時間差検出モデル、音圧差検出モデル、競合学習ネットワークを構成することで様々な聴覚情報処理を行うことができることを示した。また従来手法では個別に行っていた各手法を一つの基本素子の組み合わせだけでハードウェア上に実現することができ非常に高速かつ従来システムより少ない回路数で実現できることを示した。

音による手法は障害物の多い場所や夜間などの環境において有利であるが周囲の雑音が非常に大きい場合などには不利である。結果として画像処理

のような視覚的手段と音を用いた聴覚的手段を併用することが最適であると考えられる。視覚的・聴覚的手段を併用する場合にはそれぞれのシステムの装置規模ができるだけ小さく安価であることが望ましい。実験結果で示したように、本システムは従来手法より回路規模が小さく併用する際にも有効な方法であるといえる。また基本素子であるPNモデルの構成は先に提案した音源定位と音源種類の識別を行う回路と同じであり、識別手法毎に個別の回路を製作する必要がなく一つの装置にまとめることが容易であるため、最終的な装置の規模を小さくできる。加えて併用時には各識別結果を相互に反映させることが容易であり、性能の向上も期待できる。

本研究に関する今後の課題として、本研究では計算機を使用した周波数-パルス変換部もハードウェア上に実装し、様々な環境において実験することがあげられる。特に、実際の環境では背景雑音が非常に大きな場合や音源が複数ある場合など音環境が大きく変化するため、本論文でも述べた複数音源を分離する機構などを新たに適用して実験を行う必要があると考えられる。また先にも述べた通り、本研究の最終的な目標としては聴覚障害者への支援装置や車載用の車両検知装置などとして実用化することがある。本研究での成果が実際に製品化するための基礎となることを期待する。

謝辞

初めに、本論文を執筆するにあたり御指摘・審査を賜った名古屋工業大学大学院 岩田 彰 教授, 北村 正 教授, 梅崎 太造 教授に深く感謝致します。

そして本研究の進行および本論文の作成にあたり、多大な御尽力を頂き、終始熱心な御指導と御教示を賜った黒柳 奨 准教授, Mauricio Kugler 助教に深く感謝致します。また本研究の実現に多大な御尽力を頂いた関東学院大学 工学部電気電子情報工学科講師 元木 誠 先生, (株)シャープ 村橋 善光 様に感謝致します。

本研究の一部は財団法人堀情報科学振興財団, 文部科学省科学研究費・若手研究(B), (株)トヨタ自動車, および岐阜県知的クラスター創生事業「ロボティック先端医療クラスター」による支援を受けました。記して感謝致します。

本研究での録音やデータの収集に対して御協力を頂きました名古屋工業大学情報工学専攻岩田研究室の学生, 卒業生の皆様に感謝致します。また日頃より共に研究を行い助力を頂いている名古屋工業大学大学院 白石 善明 准教授をはじめとする白石研究室の方々に感謝いたします。

最後に名古屋工業大学への進学より5年にのぼる長い学生生活を大きな事故や病気もなく成し遂げることができたのは、ひとえに家族の支援と協力のおかげです。末筆ながら家族に謝辞を述べるとともに本論文の締めくくりとさせていただきます。

本論文著者を支援して下さった方々(敬称略)

名古屋工業大学大学院 教授	岩田 彰		
名古屋工業大学大学院 准教授	黒柳 奨		
名古屋工業大学大学院 助教	Mauricio Kugler		
関東学院大学工学部 講師	元木 誠		
シャープ株式会社	村橋 善光		
名古屋工業大学大学院 教授	北村 正	梅崎 太造	松尾 啓志
名古屋工業大学大学院 准教授	白石 善明	須藤 正時	
理研産業株式会社	鈴木正己		
株式会社 イーエスピー企画	江崎 雅康		
株式会社 アールジェイ	石田 淳一	角谷 充彦	
トヨタ自動車株式会社	宮治 正廣		
株式会社トヨタ IT 開発センター	段野 幹男		
アイシン精機株式会社	秋田 時彦	中村 友之	
財団法人 岐阜県研究開発財団			
知的クラスター本部	西村 賢	柳瀬 秀治	瀧口 昭彦
財団法人 堀情報科学振興財団	堀 誠	池谷 大介	
いいだ特許事務所	飯田 昭夫	江間 路子	
岐阜工業高等専門学校 准教授	出口 利憲		
岩田研究室 事務・元事務	原田 理絵	浅田 里香	山田 由美子

岩田研究室・松尾研究室・白石研究室 卒業生・在学生

山田 樹一	山中 幸治	吉野 栄二	露梨 真史	刑部 努	藤田 和幸
篠田 和幹	田島 隆行	平田 浩一	間瀬 友裕	鷺沢 嘉一	森本 寛
青木 一真	黄 笛	後藤 青竜	吉田 大成	加藤 一民	佐藤 雅幸
高橋 健一	川端 祥	胡 欣	伊藤 健二	八重倉 智	羽場 裕介
林 一幸	松崎 恵介	奥井 広高	小瀬 龍太	坪田 啓太	福井 宏紀
杉浦 健太郎	島 隆二	辻村 幸弘	櫻井 幹大	谷口 和雅	藤角 岳史
宮谷 俊行	川瀬 悠	大隅 春彦	美澤 匠久守	岩田 元	江尻 敬人
井上 英明	鈴木 彰浩	星野 由行	石田 光一	小出 哲久	坂本 和俊
三宅 猛	齊藤 翔	森 万純	鈴木 貴史	稲井 俊介	岩田 晁尚
宅間 勇太	安藤 貴臣	木村 明広	内海 嘉宏	山田 美穂	興梠 文佳
Anto Satriyo Nugroho	Victor Alberto Parciannelo Benso				

その他岩田研究室・松尾研究室・白石研究室 卒業生・在学生

平成 13 年度・平成 14 年度 岐阜工業高等専門学校 卒業生

長江 吉彦	高木 潤	光上 英明	小林 晋士	早崎 大地	和田 純平
石垣 薫	辻 巧美	水野 直人	岩崎 真也	岩田 和久	森 裕紀

その他 大勢の方々

参考文献

- [1] Pickles J.O.: “An Introduction to the Physiology of Hearing” , ACADEMIC PRESS, 1988
- [2] ブラスウェルト著、森本 政之・後藤 敏幸訳 : “空間音響”, 鹿島出版会, 1986.
- [3] P. Julian, A.G. Andreou,L. Riddle,S. Shamma, D.H. Goldberg,G. Cauwenberghs : “A comparative study of sound localization algorithms for energy aware sensor network nodes”, IEEE Trans. Circuits and Systems I, vol.51, Issue 4, pp.640-648, April 2004
- [4] M.S. Brandstein, J.E. Adcock, and H.F. Silverman : “A closed-form location estimator for use with room environment microphone arrays”, IEEE Trans. Speech Audio Process., vol.5, no.1, pp.45-60, 1997.
- [5] 西浦 敬信, 西岡 良典, 山田 武志, 中村 哲, 鹿野 清宏 : “CSP 法による音源位置同定を備えたマルチビームフォーミング”, 電子情報通信学会論文誌 D-II, Vol.J83-DII, No.7, pp.1610-1619, 2000.
- [6] 小林 和則, 古家 賢一, 片岡 章俊 : “位置が未知である複数のマイクロホンを用いたブラインド音源位置推定”, 電子情報通信学会論文誌 A, Vol.J86-A, No.6, pp.619-627, 2003.
- [7] R.E.Irie : “Multimodal sensory integration for localization in a humanoid robot”, IJCAI-97, pp55-59, 1990.
- [8] K.Nakadai, H.G.Okuno, H.Kitano : “Real-time Sound Source Localization and Separation fo Robot Audition”, ICSLP-2002, pp.193-196, 2002.

- [9] C.Couvreur : “Environmental Sound Recognition : a Statistical Approach” , PhD thesis, Faculté Polytechnique de Mons, Belgium, June 1997.
- [10] K.Kashino, K.Nakadai, T.Kinoshita, H.Tanaka : “Organization of Hierarchical Perceptual Sounds : Music Scene Analysis with Autonomous Processing Modules and a Quantitative Information Integration Mechanism”, In Proceeding of the 1995 IJCAI, 1995.
- [11] A.Waibel, T.Hanazawa, G.Hinton, K.Shikano, and K.J.Lang : “Phoneme recognition using time-delay neural networks”, IEEE Trans. Acousti., Speech,& Signal Process., vol.37, no.3, pp.328-339, 1989.
- [12] 奥村 元, 趙 國, 西浦 敬信, 山下 洋一 : “分散マイクロホンシステムを用いた音源位置同定”, 信学技報 SP2006-139, pp.61-66, 2007.
- [13] J.C.Segura, A. de la Torre, M.C. Benitez, and A.M. Peinado : “Model-based compensation of the additive noise for continuous speech recognition. Experiments using AURORA II database and tasks”, Proceeding of Eurospeech 2001, vol.1, pp.221-224, 2001.
- [14] 李 晃伸 : “大語彙連続音声認識エンジン Julius の開発の進展”, 信学技報 SP2005-111, pp.37-42, 2005.
- [15] Maass W., and Bishop C.M., : “Pulsed Neural Networks”, MIT Press , 1998
- [16] 黒柳 奨 : “パルス伝達型ニューロンモデルによる聴覚神経回路モデル”, 名古屋工業大学博士論文, 1996.
- [17] 田中 愛久, 黒柳 奨, 岩田 彰 : “FPGA のためのニューラルネットワークのハードウェア化手法”, 信学技報 NC2000-179, pp.175-182, 2001.
- [18] 二俣 宣義, 黒柳 奨, 岩田 彰 : “FPGA のためのパルスニューロンモデルの実装手法”, 信学技報 NC2001-211, pp.121-128, 2002.
- [19] 江口 祐俊, 古田 俊之, 堀口 浩幸, 樗木 杉高 : “学習機能をもつパルス密度形ニューロンモデルとそのハードウェア”, 電子情報通信学会論文誌 C-II, Vol.J74-CII, No.5, pp.369-376, 1991.

- [20] Hirai, Y. : “A 1,000-Neuron System with One Million 7-bit Physical Interconnections”, Advances in Neural Information Processing System 10, pp.705-711, 1998.
- [21] 村橋 善光, 掘田 博久, 道木 慎二, 大熊 繁 : “GHA 学習則を組み込んだ $\Delta\Sigma$ 変調に基づくパルスニューラルネットワークとそのハードウェア実装”, 電子情報通信学会論文誌 D-II, Vol.J87-DII, No.2, pp.705-715, 2004.
- [22] 三上廉司 : “FPGA の基礎知識”, Design Wave Magazine, pp.32-34, Jul, 2007.
- [23] 黒柳 奨, 岩田 彰 : “聴覚情報処理システムのためのパルスニューロンモデルを用いた競合学習ネットワーク”, 電子情報通信学会論文誌 D-II, Vol.J87-DII, No.7, pp.1496-1504, 2004.
- [24] 久野 宗, 三品昌美 : “脳・神経の科学 I”, 岩波書店, 1998.
- [25] 甘利 俊一, 外山 敬介 編 : “脳科学大事典”, 朝倉書店, 2000.
- [26] 川人光男, 行場次郎, 藤田一朗, 乾敏郎, 力丸裕 : “視覚と聴覚”, 岩波書店, 1994.
- [27] 境 久雄, 中山 剛, 日本音響学会 : “聴覚と音響心理”, コロナ社, 1978.
- [28] Jeffress L.A. :”A place theory of sound localization”, J.Comp.Physiol.Psychol., 41, pp.35-39, 1948.
- [29] 平田 浩一 : “パルスニューラルネットワークを用いた自己組織化マップ”, 名古屋工業大学卒業論文, 2002.
- [30] 黒柳 奨, 岩田 彰 : “音源方向定位聴覚神経系モデルによる ITD,ILD の脳内マッピングの実現”, 電子情報通信学会論文誌 D-II, Vol.J79-DII, No.2, pp.267-276, 1996.
- [31] 駒木根隆士, 平早達也 : “蝸牛の周波数分析機能を模擬するフィルターバンクの一構成法”, 信学技報 SP87-45, pp.65-71.
- [32] 三谷 政昭 : “デジタルフィルタデザイン”, 昭見堂, 1987.

- [33] 岩佐 要, 元木 誠, 黒柳 奨, 岩田 彰 : “パルスニューロンモデルによる時間差検出ネットワークの改良”, 信学技報 NC2005-150, pp.151-156, 2006.
- [34] Kohonen, T. : “Self-Organizing Maps”, Springer-Verlag, 1995.
- [35] 黒柳 奨, 岩田 彰 : “パルスニューロンモデルのための教師あり学習則”, 信学技報 NC97-151, pp.95-102, 1998.
- [36] 黒柳 奨, 岩田 彰 : “聴覚情報処理のための競合学習パルスニューラルネットワーク CONP-II”, 電子情報通信学会論文誌 D-II, Vol.J90-D, No.1, pp.115-125, 2007.
- [37] 岩佐 要, 黒柳 奨, 岩田 彰 : “FPGA を用いたパルスニューロンモデルによる音源定位及び音源種類の識別システムの実装”, 電子通信情報学会論文誌 D, Vol.J90-D, No.11, pp.3079-3090, 2007.
- [38] Iwasa, K., Kuroyanagi, S., Iwata, A. : “A Sound Localization and Recognition System using Pulsed Neural Networks on FPGA”, to appear in : Proceeding of International Joint Conference of Neural Networks 2007, August, 2007.
- [39] 深山 正幸, 北川 章夫, 秋田 純一, 鈴木 正國 : “HDLによるVLSI設計 - VerilogHDLとVHDLによるCPU設計-”, 共立出版社, 1999.
- [40] 長谷川 祐恭 : “VHDLによるハードウェア設計入門”, CQ出版社, 1995.
- [41] 並木 秀明, 後閑 哲也, 片岡 忠士 : “よくわかるSystemCによるシステムデザイン入門”, 技術評論社, 2005.
- [42] T.Nishiura, T.Yamada, S.Nakamura, and K.Shikano : “Localization of multiple sound sources based on a CSP analysis with a microphone array”, IEEE ICASSP SP-L4-6, pp.1053-1056, June 2000.
- [43] J.Valin, F.Michaud, J.Rouat, D.Létourneau : “Robust Sound Source Localization Using a Microphone Array on a Mobile Robot”, Proceedings of Intelligent Robots and Systems 2003, IROS2003, Vol.2, pp.1228-1233, Oct. 2003.

- [44] 木下 智義, 坂井 修一, 田中 英彦 : “周波数成分の重なり適応処理を用いた複数楽器の音源同定処理”, 電子情報通信学会論文誌 D-II, Vol.J83-DII, No.4, pp.1073-1081, 2007.
- [45] 佐藤 武 : “自動車工学全書 16 自動車の安全”, 山海堂, 1980.
- [46] 大塚 紳一郎, 原 寛徳, 小沢 慎治 : “車載マイクロフォンによる緊急車両の存在と方向検知システム”, 電気学会論文誌, 産業応用部門誌, D, 124, pp.388-395, 2004.
- [47] 星野 博之 : “ドライバ支援のための走行音による接近車両検知システム”, 日本音響学会誌, 62 卷 3 号, pp.265-274, 2006.
- [48] 松本 美智子, 須藤 智, 小沢 慎治 : “交差点通行時における危険警報のための接近車両検出”, 電気学会論文誌. C, 電子・情報・システム部門誌, Vol.122, No.3, pp.512-522, 2002.
- [49] Surendra G., Osama M., Robert F.K.M, Nikolaos P.P. : “Detection and Classification of Vehicles”, IEEE Trans. ITS, Vol.3, No.1, pp.37-47, 2002.
- [50] Chieh-Chi W. Thorpe C. Thrun S. : “Online simultaneous localization and mapping with detection and tracking of moving objects: theory and results from a ground vehicle in crowded urban areas”. Proceedings of ICRA 2003, pp.842-849, 2003.