Investigations into the feasibility of digital neuromorphic signal processing circuits

This item was submitted to Loughborough University's Institutional Repository by the author.

Additional Information:


Metadata Record: https://dspace.lboro.ac.uk/2134/28189

Publisher: © Lim Seow Chuan

Rights: This work is made available according to the conditions of the Creative Commons Attribution-NonCommercial-NoDerivatives 2.5 Generic (CC BY-NC-ND 2.5) licence. Full details of this licence are available at: http://creativecommons.org/licenses/by-nc-nd/2.5/

Please cite the published version.
This item was submitted to Loughborough University as a PhD thesis by the author and is made available in the Institutional Repository (https://dspace.lboro.ac.uk/) under the following Creative Commons Licence conditions.

For the full text of this licence, please go to:
http://creativecommons.org/licenses/by-nc-nd/2.5/
Please note that fines are charged on ALL overdue items.
INVESTIGATIONS INTO THE FEASIBILITY OF DIGITAL NEUROMORPHIC SIGNAL PROCESSING CIRCUITS

by

Lim Seow Chuan

A Doctoral Thesis

Submitted in partial fulfilment of the requirements for the award of

Doctor of Philosophy of Loughborough University

May 1999

© by Lim Seow Chuan 1999
Abstract

Modelling of the mammalian auditory system is valuable in understanding perception processes and has benefits in the design of signal processing systems and human prosthetic implants. However, as models increase in complexity, traditional methods of modelling using general purpose computers become very slow. One method of overcoming this is to use electronic implementations of these models. This thesis looks into the feasibility of auditory system implementations in digital technology, through the implementation of the Four-Stage Pitch System for pitch detection in hearing proposed by Hewitt and Meddis.

It presents the first ever digital implementation of a range of fundamental components found in the auditory system. Each component is based on a digital discrete-time model which not only provides the required functionality but operates in real-time while keeping down the implementation size. Each component was simulated in VHDL and then implemented in an FPGA.

This thesis also looks into the feasibility of hardware implementation of the entire Four-Stage Pitch System by analysing the architecture of the system, performing a software simulation to verify its expected functional characteristics, and analysing its expected implementation size. The results show that it is feasible to implement a complete real-time system in a single integrated circuit or an FPGA.
Acknowledgements

I would like to first thank the Overseas Research Studentship Council for providing a substantial part of the funding necessary for this work.

Special thanks to my supervisor and mentor, Professor Simon Jones for the patience and dedication in giving me guidance and encouragement throughout the project. Special thanks also to Professor Ray Meddis of Essex University for his help and advice in this collaborative work.

I would also like to thank my colleague and partner, Robert Temple, for working together with me and helping me to formulate new ideas.

My sincere appreciation extends to past and present members of the Electronic System Design Group at Loughborough University for all their assistance. In particular, I would like to thank Julian Yeandel for teaching me how to use VHDL test-benches and for setting up the UNIX side of the test system which was extensively used for the design work.

I am also very grateful for the encouragement, support and understanding of my girlfriend, Rukmani Manicavasagar, throughout these few years.

Finally and most of all, I acknowledge that without the blessings of my Lord and friend, Jesus Christ, this thesis would never have been completed.
# Table of Contents

1. Introduction ..................................... 1
   1.1. Modelling of the Mammalian Perception System .......... 1
   1.2. Hardware Implementation .................................. 2
   1.3. Motivation ................................................. 3
   1.4. Aims and Objectives of Research ............................. 7
   1.5. Structure of Thesis ....................................... 8

2. Review and Literature Survey .................. 10
   2.1. Objectives of Review .................................... 10
   2.2. The Hearing System ...................................... 10
      2.2.1. The Auditory Periphery ................................ 10
      2.2.2. The Auditory Brainstem ............................... 13
   2.3. Auditory Neural Pre-processing .......................... 16
   2.4. Modelling of the Auditory Subsystem ....................... 18
      2.4.1. Software Modelling .................................. 18
      2.4.2. Hardware Modelling .................................. 18
   2.5. Pitch System Models ...................................... 22
      2.5.1. Four Stage Pitch System .............................. 23
   2.6. Summary .................................................. 26

3. Research Overview .................................. 27
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Objectives of Chapter</td>
<td>27</td>
</tr>
<tr>
<td>3.2</td>
<td>Selection of Research Topic</td>
<td>27</td>
</tr>
<tr>
<td>3.3</td>
<td>Objectives of Research</td>
<td>28</td>
</tr>
<tr>
<td>3.4</td>
<td>Introduction to Research Topics</td>
<td>30</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Study of Four Stage Pitch System Component Implementation</td>
<td>30</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Study of System Implementation</td>
<td>31</td>
</tr>
<tr>
<td>3.4.3</td>
<td>System Simulation Results and Analysis</td>
<td>31</td>
</tr>
<tr>
<td>3.5</td>
<td>Tools and Design Flow</td>
<td>31</td>
</tr>
<tr>
<td>4.5</td>
<td>Implementation of the Four Stage Pitch System Components</td>
<td>34</td>
</tr>
<tr>
<td>4.1</td>
<td>Objectives of Chapter</td>
<td>34</td>
</tr>
<tr>
<td>4.2</td>
<td>Basic Four Stage Pitch System Specification</td>
<td>34</td>
</tr>
<tr>
<td>4.3</td>
<td>Cochlear Mechanical Filtering</td>
<td>36</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Selection of Model</td>
<td>36</td>
</tr>
<tr>
<td>4.3.2</td>
<td>Implementation</td>
<td>37</td>
</tr>
<tr>
<td>4.3.3</td>
<td>Results and Analysis</td>
<td>38</td>
</tr>
<tr>
<td>4.4</td>
<td>Hair Cell's Mechanical to Neural Transduction</td>
<td>39</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Selection of Model</td>
<td>39</td>
</tr>
<tr>
<td>4.4.2</td>
<td>The Meddis Inner Hair Cell Model</td>
<td>40</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Implementation</td>
<td>41</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Results and Analysis</td>
<td>42</td>
</tr>
<tr>
<td>4.5</td>
<td>Auditory Nerve Fibre</td>
<td>44</td>
</tr>
<tr>
<td>4.5.1</td>
<td>Selection of Model</td>
<td>44</td>
</tr>
<tr>
<td>4.5.2</td>
<td>Implementation</td>
<td>46</td>
</tr>
<tr>
<td>4.5.3</td>
<td>Results and Analysis</td>
<td>47</td>
</tr>
<tr>
<td>4.6</td>
<td>Coincidence Cell</td>
<td>47</td>
</tr>
<tr>
<td>4.6.1</td>
<td>Selection of Model</td>
<td>48</td>
</tr>
<tr>
<td>4.6.2</td>
<td>Implementation</td>
<td>48</td>
</tr>
<tr>
<td>4.6.3</td>
<td>Results and Analysis</td>
<td>49</td>
</tr>
<tr>
<td>4.7</td>
<td>Summary</td>
<td>50</td>
</tr>
</tbody>
</table>
5. Implementation of the Stellate Cell .......... 51

5.1. Objectives of Chapter ............................................. 51

5.2. Selection of Model .................................................. 51

5.3. Dendrite ................................................................. 53

5.3.1 Low Pass Filter Model ............................................. 53
5.3.2 Implementation ....................................................... 53
5.3.3 Results & Analysis .................................................. 54

5.4. Stellate Cell Soma ..................................................... 55

5.4.1 Cell Soma Model ..................................................... 55
5.4.2 Implementation ....................................................... 60
5.4.3 Results & Analysis .................................................. 61

5.5. Cell Soma Resolution Experimentation ....................... 64

5.5.1 Motivation and Experimentation Setup ....................... 64
5.5.2 Experimentation Results and Analysis ....................... 64

5.6. Summary ................................................................. 68

6. Pitch System Implementation ................................. 70

6.1. Objectives of Chapter ............................................. 70

6.2. System Architecture .................................................. 70

6.2.1 Architecture Based on a One to One Approach ............... 70
6.2.2 Sharing the Noise and Dendrite Units ......................... 72

6.3. Applying Multiplexing ............................................... 74

6.3.1 One of Each Component Multiplexing Approach .......... 74
6.3.2 Multi-channel Approach to the Whole System ............. 75
6.3.3 Single Cochlear Channel Approach ......................... 79

6.4. Implementation ......................................................... 81

6.4.1 VHDL and FPGA Implementations ............................ 81
6.4.2 Labview based Implementation ............................... 82

6.5. Summary ................................................................. 82
## Table of Contents

7. **Pitch System Test and Analysis** .......... 84

7.1. Objectives of Chapter ................................. 84

7.2. Test System Setup ................................. 84

7.3. Simulation Results and Analysis ....................... 86

7.3.1 Pure Tone Test ........................................... 86
7.3.2 Random Noise Test ................................. 88
7.3.3 Modulated Tone Test ................................. 91
7.3.4 Modulated Tone Intensity Test ........................ 94

7.4. Further Analysis of Results .......................... 96

7.5. Conclusion ..................................... 97

8. **Conclusion** ........................................... 99

8.1. Objectives of Chapter ................................. 99

8.2. Review of Objectives .................................. 99

8.3. Conclusions ........................................... 101

8.4. Measure of Success ..................................... 102

8.5. Limitation of Work ..................................... 103

8.6. Suggestions for future investigations .................. 104

8.6.1 Extensions of Current Investigations ................. 105
8.6.2 Related Investigations ................................. 106

References ................................................. 107

Appendix: Publications
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>Structure of the auditory periphery.</td>
<td>11</td>
</tr>
<tr>
<td>2-2</td>
<td>Cross-section of the cochlea duct.</td>
<td>12</td>
</tr>
<tr>
<td>2-3</td>
<td>Neural cell structure.</td>
<td>14</td>
</tr>
<tr>
<td>2-4</td>
<td>Membrane voltage and ionic currents during the generation of a spike.</td>
<td>15</td>
</tr>
<tr>
<td>2-5</td>
<td>Simplified organisation of the auditory brainstem.</td>
<td>16</td>
</tr>
<tr>
<td>2-6</td>
<td>Structure of Lyon and Mead serial cascade of second-order filters cochlea implementation.</td>
<td>20</td>
</tr>
<tr>
<td>2-7</td>
<td>Data flow structure of the cochlea and a single filter used in Summerfield and Lyon digital cochlea implementation.</td>
<td>22</td>
</tr>
<tr>
<td>2-8</td>
<td>The Four Stage Model for pitch detection.</td>
<td>24</td>
</tr>
<tr>
<td>3-1</td>
<td>Standardised design flow.</td>
<td>32</td>
</tr>
<tr>
<td>3-2</td>
<td>Hardware set-up.</td>
<td>32</td>
</tr>
<tr>
<td>3-3</td>
<td>Close-up of the test board with the Xilinx XC4025PG223-6 FPGA.</td>
<td>33</td>
</tr>
<tr>
<td>4-1</td>
<td>Data flow diagram of a second order Butterworth IIR bandpass filter.</td>
<td>37</td>
</tr>
<tr>
<td>4-2</td>
<td>Architecture of a cochlear bandpass filter unit.</td>
<td>38</td>
</tr>
<tr>
<td>4-3</td>
<td>Response of the cochlear bandpass filter with centre frequency of 3350Hz to random noise.</td>
<td>38</td>
</tr>
<tr>
<td>4-4</td>
<td>Meddis inner hair cell model equations.</td>
<td>40</td>
</tr>
<tr>
<td>4-5</td>
<td>Flow diagram of the Meddis inner hair cell model.</td>
<td>41</td>
</tr>
<tr>
<td>4-6</td>
<td>Data flow structure of the inner hair cell implementation by Temple.</td>
<td>42</td>
</tr>
<tr>
<td>4-7</td>
<td>Response of the Meddis hair cell implementation to step intensity sinusoid.</td>
<td>42</td>
</tr>
<tr>
<td>4-8</td>
<td>Response of the hardware Meddis hair cell model to a 30 dB, 30% modulated sinusoid.</td>
<td>43</td>
</tr>
<tr>
<td>4-9</td>
<td>Diagram of auditory nerve interconnections between hair cells and stellate cells.</td>
<td>44</td>
</tr>
<tr>
<td>4-10</td>
<td>Diagram of simplified method of simulating nerve fibres.</td>
<td>45</td>
</tr>
<tr>
<td>4-11</td>
<td>Noise scaling functions, S(p).</td>
<td>46</td>
</tr>
<tr>
<td>4-12</td>
<td>Architecture of the auditory nerve fibres unit implementation.</td>
<td>46</td>
</tr>
<tr>
<td>4-13</td>
<td>Results of the auditory nerve fibres unit when stimulated by the output of the inner hair cell unit.</td>
<td>47</td>
</tr>
<tr>
<td>4-14</td>
<td>Data flow diagram of the coincidence cell model.</td>
<td>48</td>
</tr>
<tr>
<td>4-15</td>
<td>Simplified architecture of the coincidence cell unit.</td>
<td>48</td>
</tr>
<tr>
<td>4-16</td>
<td>Input and output waveforms of the coincidence cell.</td>
<td>49</td>
</tr>
<tr>
<td>5-1</td>
<td>MacGregor’s compartmentalised model.</td>
<td>52</td>
</tr>
<tr>
<td>5-2</td>
<td>Data flow diagram of a 1\textsuperscript{st} order butterworth IIR lowpass filter.</td>
<td>53</td>
</tr>
<tr>
<td>5-3</td>
<td>Architecture of a dendrite lowpass filter.</td>
<td>54</td>
</tr>
<tr>
<td>5-4</td>
<td>Response of the dendrite lowpass filter to the output from the auditory nerve</td>
<td></td>
</tr>
</tbody>
</table>
fibre unit.

Figure 5-6 PTRN10 model equations 55
Figure 5-5 State variable behaviours of the point neuron model. 56
Figure 5-7 Equation for approximating the output spiking potential. 56
Figure 5-8 Equation for membrane resistance. 57
Figure 5-9 Discrete integration implementation of the threshold equation. 57
Figure 5-10 Modified Gk equation. 58
Figure 5-11(a) Actual neuronal response to depolarising current of 0.6 nA 59
Figure 5-11(b) Stellate cell-soma model response plotted with reference to cell's resting potential. 59
Figure 5-11(c) Stellate cell-soma model response with threshold set to a constant. 60
Figure 5-12 Data flow diagram of the stellate cell soma unit. 60
Figure 5-13 Architecture of stellate cell soma implementation. 61
Figure 5-14 Response of stellate cell soma implementation to 20 mV stimulus. 62
Figure 5-15 Stellate cell soma responses to the output from the dendrite filter. 63
Figure 5-16 Sum of the outputs of 40 stellate cells with different BMF when stimulated by the same modulated pure tone. 63
Figure 5-17 Average pulse period error in relation to Gk and E fractional bit resolution and the epoch period, dt. 66
Figure 6-1(a) Top level data flow architecture of the FSPS. 71
Figure 6-1(b) Architecture of a cochlear channel based on one to one approach. 71
Figure 6-2 Data flow structure of a cochlear channel where auditory nerve fibre units and dendrite filter units are shared. 73
Figure 6-3 System layout using one of each component to simulate all components in the system through multiplexing. 74
Figure 6-4 System layout using multi-channel multiplexing scheme. 75
Figure 6-5 Data flow architecture of a single cochlear channel of the Four Stage Pitch System with multiplexing. 79
Figure 6-6 Hardware structure of the one cochlear channel implementation. 79
Figure 7-1 Typical plot of the simulation outputs. 85
Figure 7-2 Pure tone input test at different frequencies and intensity. 86
Figure 7-3 Response to random noise input with various intensities and cochlear filter centre frequencies. 89
Figure 7-4 Response to 30dB modulated pure tones. 91
Figure 7-5 Representative results of amplitude modulated intensity test. 95
List of Tables

| Table 4-1 | Summary of speed and size of components in this chapter | 50 |
| Table 5-1 | Parameter values for the stellate cell soma model. | 58 |
| Table 5-2 | Resolution selection for different accuracy requirements for the stellate cell soma. | 68 |
| Table 5-3 | Summary of speed and size of all components. | 68 |
| Table 6-1 | Estimated system size based on one to one approach. | 72 |
| Table 6-2 | Estimated system size with sharing of the dendrite filter units and the auditory nerve fibre units. | 73 |
| Table 6-3 | Number of channels verses the minimum required epoch and clock rate of the components to achieve real-time operation | 76 |
| Table 6-4 | Number of channels in the system verses the system size | 76 |
| Table 6-5 | Memory requirements of each type of component and for the entire system. | 77 |
Chapter One

Introduction

1.1 Modelling of the Mammalian Perception System

This thesis looks into the use of digital electronic technology as a means of auditory system modelling. Modelling of the sensory mechanical and neural processes has become essential in the study of human sensory perception. This is primarily for the following reasons:

- Modelling is necessary since direct physical investigations and measurements of animals are limited. Physiologists are only able to stimulate and measure the response of a small number of neural cells at a time. This limited view of the system can only help to provide an understanding of the functions of individual, and small groups of cells, providing clues to the operation of the entire system. These clues often result in different and sometimes conflicting theories of perception. Modelling can provide a way to understanding large neural systems by first simulating the individual cells, and then building large systems where theories can be tested and refined.

- The public’s viewpoint has in recent years become increasingly hostile towards experimentation on animals. As a result, there is a drive towards reducing the need for these experiments as far as possible, even to the extent of replacing them with alternatives. One of the most effective ways of working towards this goal is to use modelling. Modelling can provide information about the operation of the neural system and help pinpoint areas where physiological studies are needed. As more
information is gleaned from physiological studies to improve the models, the need for invasive animal studies will be reduced.

- There is also an increasing interest in the use of biological models in the area of signal processing. It is clear that the mammalian brain has a signal processing capability that far exceeds the capability of algorithmic techniques, which is achieved using relatively slow and inaccurate components. Speed in the mammalian brain is achieved using massive parallelism as opposed to standard serialised computing techniques. Although artificial neural networks like multi-layer perceptrons (MLP) and many others exist, they tend to take a very simplistic abstraction of the brain. They also ignore the fact that the brain, especially for sensory processing, has complex structures made up of many types of neurons as opposed to a homogeneous network of similar cells. Such artificial neural networks are generally unable to capture the full extent of the capability of the brain. Thus the idea of copying the brain by following more closely the way the brain functions will become ever more attractive. However, in order to do that, a good understanding of the neural processes is needed. This understanding will be contributed in part through modelling.

1.2 Hardware Implementation

By far the most common method of modelling biological neural systems is to use some form of computer, ranging from a PC to a top of the range workstation. Inevitably as models increase in size and complexity, the computing power of these computing devices is highly stretched. Simulations are often very slow, usually at a small fraction of the speed required for real-time operation. As a result, modellers are increasingly looking into the use of other alternatives to speed this up. These include the use of multi-processor general purpose supercomputers, hardware accelerators and even electronic hardware implementations of the models themselves.
Chapter I Introduction

The rapid development of semiconductor technology has resulted in tremendous increases in the circuit density of integrated circuits. It is now possible to implement very large circuits containing millions of transistors or logic gates onto a single silicon chip. Therefore, the option of implementing large system models directly onto hardware becomes viable, allowing modellers to emulate in real-time and at a reasonable cost. These implementations will not only benefit sensory modellers, but can have potential benefits in other areas. One such area is the design of signal processing where the experience gained can greatly influence and improve the design of the front-end of vision and speech processors. The electronic implementations themselves can also potentially be applicable to human implants, improving on existing devices like cochlear implants and future retina implants.

This thesis looks into the electronic implementation of models of the initial stages of the mammalian auditory system, from the outer ear to the auditory cortex. The auditory system is chosen because of the clearer knowledge of the structure, function and purpose of the cells and brain stem structure that exists through extensive physiological studies carried out in the past. Another reason for this choice is the 2-dimensional nature of the stimulus (amplitude and time). Such a system is much easier to control and its structure is simpler to study and design compared to 4 or more dimensional systems such as vision or olfaction. Even though the auditory system is simpler than the other systems, the knowledge gained through its implementation is applicable to other sensory system modelling. This is because the underlining problem of abstraction of biology into hardware is common to all sensory models.

1.3 Motivation

The electronic implementation of the auditory system model can be divided into two main areas. One area focuses on the use of analogue VLSI while the other employs various forms of digital implementation technology. Presently, most of the work done in this field uses analogue VLSI with relatively very little work done in digital VLSI. This is primarily because analogue implementations have some advantages that
benefit the electronic implementation of neuromorphic systems. These advantages are as follows:

- Analogue implementations operate in real-time and in a continuous way.
- Due to the nature of transistors, with their linear and logarithmic (with sub-threshold biasing) characteristics, it is easier to implement simplified models of the neurons in analogue VLSI. These neurons are usually implemented using only a few transistors.
- Using a one-to-one approach, directly mapping each cell to an analogue circuit unit, the system architecture is much more straightforward to design.
- Designers can use sub-threshold biasing techniques in their circuits to achieve very low-power usage.

However, there are disadvantages in the use of analogue VLSI and these impose serious limitations during design and use. These are as follows:

- Analogue implementations are inflexible, i.e. they usually cannot be programmed. Using external components to bias circuits is possible. However, when large number of units need to be biased, structures like shared biasing or ladder networks must be used, which again reduce flexibility and increase system implementation size.
- Communication on a silicon chip is usually two-dimensional. Even when multi-layer wires are used on silicon, these are still limited to a handful of layers. Therefore, it is difficult to implement the massive number of connections, as in the brain, on a silicon wafer. Often, designers have to resort to techniques like multiplexing, switching networks or coding into and from digital, which results in some loss of information or continuity and also significantly increases the system implementation size.
- There is a need to have a one-to-one, direct mapping approach in the design of a system in analogue VLSI. As a result, the high speed inherent in silicon implementations is not utilised. This is due to the difficulty of multiplexing
analogue circuits caused by the limitations of on chip storage elements based on capacitors and switches.

• Implementation of analogue circuits usually requires a number of design iterations. Generally, to verify whether an analogue design works, the circuits have to be implemented in silicon and tested. Each cycle usually takes around three to six months from the submission of the design to the fabricator to receiving the fabricated silicon chips. Simulations of analogue circuits cannot usually provide an accurate prediction of its correctness, therefore designers run a high risk of chips not working after implementation in silicon.

• Analogue VLSI circuits are temperature sensitive. This is especially true for sub-threshold circuits. Such circuits are therefore difficult to calibrate and are inherently unstable. The performance of such an implementation will depend heavily on the environment it is in.

• Inaccuracy during implementation means that it is difficult to create multiple transistors or circuits with exactly the same characteristics on the same silicon wafer. Therefore, calibration has to be performed on individual circuits.

• Analogue circuits are inherently more difficult to test, especially when a high level of integration is used.

These disadvantages can be compared with the advantages of digital implementations as follows:

• Digital implementations, through the use of registers to store configuration information, offer far better flexibility, especially when large numbers of programmable units are required in a single integrated circuit.

• By employing multiplexing techniques, much higher connectivity can be achieved between circuit elements using the same number of physical wires.

• Multiplexing of circuit modules, which is easier in digital circuits, allows a few fast implementations to emulate many slow units in real-time, helping to reduce system size.

• The advent of Hardware Description Languages (HDL) like VHDL and synthesis tools allows designs to be generated quickly. Chips like Field Programmable Gate
Arrays (FPGA) can be used many times to verify circuits before the final implementation onto an Application Specific Integrated Circuit (ASIC), thus reducing the number of design iteration cycles time and the design cost.

- Digital circuits are less temperature sensitive and inherently stable. As a result, the performance is more predictable in varying environmental conditions.
- Accuracy in digital systems can be easily tailored to requirements. A common method in design for increasing accuracy is to provide more resolution by using a higher bit-width.
- Digital logic circuits are easier to test. This is the result of having a more predictable and defined system where circuit operating characteristics can be accurately predicted using software simulations.

The advantages of digital over analogue implementations are reflected in the commercial world where analogue systems are now increasingly being superseded by equivalent digital implementations. Where circuits were once predominantly analogue, they are being replaced by digital circuits with an analogue front and back end to precondition and convert them into and from digital. This is contrary to auditory system implementation where analogue systems predominate. It is therefore appropriate that a digital approach be investigated for its feasibility, especially for large models of the auditory system.

However, using digital VLSI as a means of implementing neuromorphic systems does have disadvantages. These have to be addressed to make it feasible for this approach to be used effectively. These disadvantages are as follows:

- Equivalent digital discrete models of the components are different and usually more complex than analogue versions. Therefore, simple and efficient abstractions of each component in the system have to be found that will model their functionality sufficiently enough to capture their important behaviours.
- Based on a one to one implementation of a typical neural cell, a digital cell will always be very much larger than an equivalent analogue cell. There is a need to investigate ways to compensate for this.
Chapter 1

Introduction

• Noise is essential to the operation of many neural or biologically based systems. Since it is harder to generate un-correlated noise in digital circuits, ways of reducing the complexity of noise generators and applying them efficiently has to be found.

• Because of quantisation, and limits on dynamic range in the digital representation of analogue values, the issue of accuracy and resolution have to be investigated. This is needed in part to avoid over and under flow affecting the system drastically.

How well we tackle these points will determine the feasibility of using digital implementation as an approach to the modelling of the auditory system.

1.4 Aims and Objectives of Research

Having made clear the motivations for the digital implementation of the auditory brainstem model, the aim of this area of work is therefore to understand how the biological auditory sensory processing system can be modelled efficiently in hardware. This aim can be divided into two points:

• To understand how much of the biological system structure, complexity and components are needed to capture effectively the capabilities of biological systems.

• To know how the important neuromorphic components and systems can be implemented efficiently in digital hardware.

In line with these aims and motivation, the two main objectives of this thesis are:

• To demonstrate the feasibility of implementing auditory brainstem models in real-time digital hardware.

• To identify implementation issues that are critical to such systems and to propose ways of dealing with them.
These objectives will be achieved by implementing the Four-Stage Pitch System (FSPS) suggested by Hewitt and Meddis [Hewitt94]. The system models the pitch detection capability found in the auditory brainstem. It is based on physiological studies that found neurons in the cochlear nucleus and in the inferior colliculus of the auditory brainstem capable of producing spikes in synchrony with the modulation of the stimulus. Since the modulation itself can be perceived as pitch, the detection of modulation is also referred to as pitch detection for this system.

The FSPS will provide a good test bed since it is sufficiently large to make computer simulations slow. It also includes representative components from the auditory periphery through to the auditory brainstem, typical in many models of this type.

1.5 Structure of Thesis

After this introductory chapter, Chapter 2 will look into the background needed for the appreciation of this work. This includes a simplified background on the auditory system and some related work on the electronic implementation of the auditory brainstem. This chapter will also expound on the FSPS that forms the basis of the work in this thesis.

Chapter 3 will then look further into the objectives of this work and also detail the approach and experimentation that need to be carried out in achieving these objectives. This chapter will also give a brief description of the tools and design methodology used in this research.

Next, in Chapter 4, the implementation of most of the components of the Four Stage Pitch System will be investigated. It will include the respective digital models used and the implementation results.

The stellate cell implementation will be detailed in Chapter 5. This component is presented separately since it is the most numerous component in the system. This
Chapter 1 Introduction

Chapter will include details on how the model was simplified and also how the resolution required by the model to function properly was determined.

Chapter 6 will look into the implementation of the entire Four Stage Pitch System. In it, we describe how we arrive at the final architecture of the system. This includes simplifications to the structure and the application of multiplexing to the system to reduce its overall size.

Chapter 7 then looks into the simulation of the design to determine how close its functionality corresponds to the expected results. It will also attempt to discover the reasons when the implementation fails to meet expectations.

Finally, Chapter 8 concludes by examining how well the work was able to fulfil the objectives laid out initially. In this chapter, we also propose future investigations that can be carried out on the current work and in related fields.
Chapter Two
Review and Literature Survey

2.1 Objectives of Review

This chapter presents the relevant background of the work in this thesis. The objectives here are:

• To review briefly the relevant background on auditory physiology and perception.
• To review past relevant work that has been carried out on the electronic implementation of the auditory system.
• To place the work pursued in this thesis in context with related works.
• To review the Four Stage Pitch System which is the basis of the work in this thesis.

2.2 The Hearing System

The auditory system is divided into two parts, the auditory periphery and the auditory brainstem. The input sound stimulus is first processed by the auditory periphery mainly using mechanical structures. It is then processed by the brainstem using a large network of neurons before being fed to the auditory cortex.

2.2.1 The Auditory Periphery

Figure 2-1 shows the structure of the auditory periphery. The auditory periphery system can be subdivided into three parts. The outer ear, middle ear and inner ear.
The Outer Ear

The outer ear consists of the pinna, concha, and the ear canal. These outer ear structures are used for gathering sound and they also act as filters and resonators, modifying the sound as it reaches the eardrum. These together have the effect of increasing the sound pressure, especially within the frequency range of 2kHz to 7kHz. Most of this is caused mainly by the resonating effect of the concha and the ear canal. The pinna and concha also help in the process of sound localisation by modifying the frequency intensity response and timing differences between the two ears.

The Middle Ear

The middle ear consists of the ear drum (tympanic membrane) and three ossicles, (called the malleus, incus, and staples) with a few ligaments and small muscles that hold the ossicles in position. The purpose of such an arrangement is to match the impedance between the air medium and the much denser fluid medium of the inner ear. Much of this matching is provided by the immense surface area difference between the tympanic membrane and the footplate of the staples. The levering action of the ossicles also provides some matching, but to a lesser degree. Muscles on the
ossicles provide a limited amount of damping during the sound transfer, especially at low frequency and on onset of loud sounds [Pickles88, Chapter 2].

The Inner Ear

Vibration of the ossicles is coupled into the cochlea through the oval window. The fluid filled cochlea is made up of a coil-like boney cavity and is subdivided along its length by two tissues, the basilar membrane and Reissner's membrane. The subdivisions formed are called the scala vestibuli, scala media and scala tympani. At one end of the cavity (apex), the scala vestibuli and scala tympani are connected. Figure 2-2 shows a cross-section of the cochlea duct.

On the basilar membrane, there are hair cells that convert mechanical movements of the basilar membrane into neural spikes. The thickness of the basilar membrane changes from thick with a narrow width near the staples to thin and wide at the apex. When the cochlea is stimulated by sound, the basilar membrane will vibrate. As a result of the change in thickness and width along the cochlea duct, the vibration will then appear to form a traveling wave starting near the staples, traveling down towards the apex, constantly slowing down along the way. Also because of the varying thickness and width of the membrane, the envelope of the traveling wave will peak at locations along the length of the membrane where the natural frequency of the
membrane matches the stimulating frequency. This provides a close mechanical equivalent to the Fourier transform.

There are two types of hair cells on the basilar membrane: the outer hair cells and the inner hair cells. The outer hair cells are believed to be able to alter the tuning of the basilar membrane by changing the size of their stereocilia that touch the tectorial membrane. The inner hair cells, however, are considered to be mainly responsible for the transduction of sound into neural spikes since more than 90% of the auditory nerve fibres are connected to them. These inner hair cells, however, do not produce spikes, but instead they influence the nerve fibers connected to them by releasing neurotransmitters whenever their stereocilia are moved in a particular direction. These auditory nerve fibers then pass the information as spikes to the auditory brainstem [Pickles88, Chapter 3].

2.2.2 The Auditory Brainstem

The typical neural cell

The auditory brainstem is made up of large numbers of different types of neural cells. These cells are capable of using sudden voltage fluctuations (spikes) within their cell body not only to communicate information but also to process information. Spike (all or none) communication has inherent tolerance to noise and is able to overcome the relatively high resistance within the cell. Some examples of these neural cells are bushy cells, fusiform cells and stellate cells. Each type of cell behaves differently to its stimulus. To understand the mechanism by which such a neural cell generates spikes, there is a need to know how these voltages are generated and regulated. Figure 2-3 shows a diagram of a typical neural cell structure which can be classified into three main parts: the cell body (soma), the dendrites and the axon. Generally, all nerve cells have these main parts, but their sizes and complexity vary widely from cell to cell.
The cell soma is essential for the cell survival as it contains the nucleus. A membrane separates the contents within the cell from the exterior. As a result of different concentrations of biochemical ions like sodium and potassium ions, a potential difference exists across the membrane between the inside and outside of a cell. This difference is maintained by the cell’s internal biochemical pump actions when at rest.

On the membrane, there are molecular gates that control the diffusion of different types of ions across the membrane. When the cell membrane is stimulated externally by chemicals (neurotransmitters) emitted by other neurons it is connected to, these molecular gates will allow a larger amount of positive sodium ions to diffuse into the cell soma, resulting in a drop of potential. This influx of sodium ions is slowly balanced by the cell’s internal pump actions. However, if the decrease in potential continues, the potential will reach a threshold level where a chain reaction will be triggered, resulting in the opening of all sodium channels, and the depolarization of the cell. Once the cell has depolarized, the potassium channels will open, resulting in an outflow of potassium ions which pulls the potential back up, which will close most of the sodium gates. A spike is therefore generated. The potassium channel will slowly close again and this cycle continues as long as the input stimulation is strong enough.

Figure 2-4 shows a graph of the membrane voltage fluctuations during the generation of a spike with the flow of ions (currents). The period between nearest possible spikes of the cell is called the refractory period, which is usually no less than 1 ms.
The dendrites are tree-like structures that usually increase in size as they approach the cell body. Axons from other neurons usually make connections to the dendrites, making it the input of the cell. These connections are called synapses. Although there is no actual electrical connection, a spike arriving at the synapses will cause neurotransmitters to be released into the gap in-between called the synaptic cleft. These neurotransmitters will then open ion channels on the membrane of the dendrites resulting in polarisation or depolarisation, which in turn will propagate to the cell body. Depolarisation excites the cell towards producing spikes while polarisation usually inhibits the cell. Filtering of the impulse occurs along the way so that if the distance is long and the dendrites narrow, it will produce an effect that is identical to that of a lowpass filter. This effect will smooth the impulse and introduce delay.

The axon is a tube like structure, and it length varies from less than a fraction of a millimetre to several metres. Some are coated with a layer of fatty material called the myelin sheath. The axon transfers the spiking potential of the cell body to synapses using the same spiking process described earlier. Myelinated axons transmit these spikes faster because of lower membrane capacitance.
The Auditory Brainstem Structure

The neural cells in the brainstem are organised in groups, often in relation to their sensitivity to the stimulus frequency (tonotopical organisation). The major groups are the cochlear nucleus, superior olivary complex, inferior colliculus and the medial geniculate body. Figure 2-5 shows a simplified organisation of these groups which lead from the cochlea to the auditory cortex of the brain.

![Figure 2-5: Simplified organisation of the auditory brainstem.](image)

Physiological studies on animals have uncovered and documented many of the neural circuits, especially in the initial stages of the brainstem. These circuits includes those that runs within each grouping of cells and also longer neurons that connect different groups in the brainstem up to the cortex. Among them, researchers have the most information on the cochlea nucleus and the superior olivary complex because they are easier to measure due to their location on the auditory brainstem.

2.3 Auditory Neural Pre-processing

The discovery of well structured neural circuits led researchers to believe that a substantial amount of pre-processing of the auditory stimulus is performed in the
auditory brainstem before arriving at the cortex. A good example of such pre-processing is found in the superior olive. In the medial superior olive and lateral superior olive, there are many neural cells that receive stimulus from both the left and right cochlear nucleus. Most cells in the medial superior olive are sensitive to low frequency interaural time differences [Aitkin86] and, in the lateral superior olive, they are sensitive to interaural intensity differences at higher frequencies [Goldberg69]. This along with clearly defined circuits lead researchers to believe that the superior olive is involved in the perception of sound localisation by detecting these two important parameters.

Despite the wealth of evidence supporting the concept of auditory pre-processing in the brainstem, it is, however, difficult to appreciate fully the full extent of pre-processing and feature extraction performed because of the limitations of physiological studies. Faced with this, researchers are increasingly relying on computer modelling of the physiological system as a way of visualising and testing theories on perception.

Pitch is defined as the perceived frequency of the stimulus tone when they are separated in time [Pickles88, page 271]. For a pure tone, the pitch is perceived as the tone's fundamental frequency. But with complex tones which contains a number of frequency component (each with a frequency at a number of times that of the fundamental), even if the fundamental frequency component is missing, the pitch at the frequency of the fundamental component can still be heard. [Pickles88, page 273] For example, a complex tone with components at 1000Hz, 1100Hz and 1200Hz, can be perceived to have a pitch of 100Hz.

There are two main theories on the pitch perception; The place theory and the temporal theory (Time coding of frequency). The place theory suggests that pitch discrimination is dependent on the detection of the place of excitation of the nerve fibers along the basilar membrane [Goldstein73; Terhardt74; Wightman73]. However, physiological studies shows that at low tone frequencies of below 5 kHz, where the pitch of voice lies in, the cochlea is poor at resolving frequency [Pickles88, page 272]. This theory is also unable to explain some pitch phenomena. An example of such a
phenomena is the pitch sensation of amplitude modulated noise [Burns76]. The temporal theory states that pitch is detected by processing the periodicity of the nerve impulses which is phase locked to the stimulus, regardless of the place of excitation on the basilar membrane [Bilsen69; Licklider51; Moore82; Schouten70]. However at above 5 kHz, pitch detection with this mechanism will be poor due to the loss phase of locking characteristics of the nerve fibers at high frequencies [Rose71]. A more modern formulation accepts both theories, where the place coding is used at higher frequencies while time coding is used at frequencies below 5kHz [Pickles88, page 271].

2.4 Modelling of the Auditory Subsystem

Auditory system models can be classified into two main types: those that use software and those that use electronic hardware.

2.4.1 Software Modelling

Most modeling of the auditory subsystem is presently performed using software on general purpose workstations. This approach provides the flexibility to change and refine the model quickly and easily. A number of auditory system models exists. These include those on the auditory periphery [for example, Giguere94], individual neural cells [for example, Arle90] to those covering subsystems of the brainstem [for example, Pont91]. However, software models are often significantly slower compared to the real-time operation of their biological counterparts. This problem limits the size of the models and the real-time duration that can be simulated. In turn, it will limit the usability of this approach as the complexity of the models increase beyond the computing capability of general purpose workstations.

2.4.2 Hardware Modelling

An approach to speeding up auditory model simulations is to implement them directly in electronic hardware. However, modelling using electronic implementations result in
systems that are generally less flexible than software implementations. But they promise real-time operation, allowing experimenters to test and visualise the operation of auditory processing systems in real-time.

As mentioned in Chapter 1, the electronic hardware implementation of the auditory subsystem can be divided into two areas: Analogue VLSI and Digital VLSI. Almost all electronic neuromorphic implementations are presently based on analogue VLSI with the majority limited to the modelling of the signal processes in the auditory periphery.

**Analogue VLSI Implementations**

The earliest example of an analogue VLSI implementation of the auditory periphery is that of Lyon and Mead [Lyon88]. They used a serial cascade of second-order filters with exponentially decreasing cut-off frequency to model the cochlea as shown in Figure 2-6. This work was followed by many similar but improved implementations [Lazzaro89b; Liu92; Watts91, Watts92] including some with non-linearity [Bhadkamkar93], and those based on different analogue implementation techniques [Andreou93; Park93; Lin92]. More recent examples of cochlear implementations are that of Fregniere and Van Schaik [Fragniere97a, Fragniere97b; Schaik96a]. Even though the method of using a serial cascade of filters in VLSI implementations is commonplace, there is less use of this technique in software modelling. Software modelers, pre-dominantly use independent arrays of filters to model the cochlea in software [Cooke93; Hewitt92; Hewitt93; Hewitt94; Pont91].
Chapter 2

Review and Literature Survey

One section of the passive bidirectional model implemented in analogue VLSI

Figure 2-6: Structure of Lyon and Mead serial cascade of second-order filters cochlear implementation.

There have also been analogue implementations that model the temporal adaptation found in the transductions of sound into neural impulses in the auditory periphery. These include the work of Lazzaro [Lazzaro92a], and Van Schaik [Schaik96b]. These implementations take a high level abstraction of the physiological function by using simple saturation and adaptation circuits to model the characteristics of the hair cell and nerve fibre. While Lazzaro models the spiking capability by using a simple spike generator, Van Schaik chose instead to leave out the spike generator altogether, opting to use the output of the hair cell as inputs directly into the other neurons.

Simple neural cells that are functionally close to the biological counterparts have also been implemented in analogue VLSI. Examples of these include the work of Mahowald and Douglas [Mahowald91], Van Schaik [Schaik96b], Lazzaro [Lazzaro92b] and Mead [Mead89]. Others like Laflaquière et al [Laflaquière97] and Linares-Barranco [Barranco91], for example, implement cells that attempt to model more accurately all the essential biochemical processes in the cell. Most of these cells are designed as building blocks for future larger systems.
The work of Lazzaro and Mead is an early example of implementing the pre-processing in the auditory brainstem. They implemented a binaural localisation system [Lazzaro089a] and a pitch detection system [Lazzaro089c], both based on an array of delay lines and logical AND elements to perform a running autocorrelation of the cochlear outputs. Lazzaro and Mead's implementations however are highly abstracted versions of the brainstem sub-processing and therefore do not implement cells that behave functionally close to the biological cells. Physiological evidence supports the existence of delay in neural fibres and also found cells similar to the logical AND units for the purpose of localisation [Aitkin86]. However, the physiological evidence is pointing away from this type of system in pitch detection, and towards systems based on cells that detect amplitude modulation [Frisina90; Kim90; Hewitt92; Hewitt93; Hewitt94].

A more recent implementation of the sub-processing of the auditory brainstem is that of Van Schaik. [Schaik97] His implementation is based on the four-stage pitch system by Hewitt et al [Hewitt94] which uses the phase locking characteristics of arrays of cells to detect and lock on to amplitude modulation. Van Schaik's system was built using an analogue VLSI cochlea based on the serial cascade of filters, along with simplified neurons for the detection of modulation.

Digital VLSI Implementations

Much less work, however, is found in the area of the digital VLSI implementation of the auditory system. Most of the work in digital VLSI is focused instead on reconfigurable neural accelerators, like that of the Spike 128 System [Hartmann97], which is based on a re-configurable array of simple digital spiking neurons. Thus far, the only work in digital VLSI that implements auditory sub-systems directly in hardware is that of Summerfield and Lyon [Summerfield92]. They implemented a bit-serial digital ASIC version of a cochlea based also on a serial cascade of non-linear second order lowpass filters. However, each of the filter units incorporates an automatic gain control loop to attempt to simulate the sharp tuning characteristics of the mammalian cochlea. Figure 2-7 shows the structure of one of the filters used in their implementation.
Figure 2-7: Data flow structure of (A) the cochlea and (B) a single filter used in Summerfield and Lyon digital cochlear implementation.

As for digital VLSI implementation of higher brainstem functions, like localisation or pitch detection, none has been found thus far in this review. Because digital implementation of the auditory sub-system is almost non-existent, and given the benefits that can potentially be gained, there is a need to investigate this approach as a means of auditory system modelling.

2.5 Pitch System Models

Two main types of models exist for pitch detection in the auditory brainstem based on the detection of the periodicity of the input stimulus. The first type of models, and by far the most prevalent, [Broadbent75; Cheveigne86; Lazzaro89c; Loeb83; Lyon84; Meddis91a; Meddis91b; Moore82; Noorden82; Patterson87; Slaney90] is base on the idea of the auditory brainstem performing an auto-correlating function. The auto-correlation function is believed to be performed by neurons matching the input stimulus with one that is progressively delayed. Even though physiological evidence supports the existence of delay lines in the brainstem, they are found to be primarily
used in localisation [Aitkin86]. At present few physiological evidence is available to support auto-correlation being performed in the auditory brainstem.

The second type of model [Hewitt94] is based on the idea that periodicity, and hence pitch, is detected with many groups of neurons with each group specifically tuned to detect one pitch. This is well supported by physiological studies which shows that many cells in the cochlear nucleus and inferior colliculus have the ability to detect and amplify amplitude modulation. It also was found that cells in the inferior colliculus are organised according to the frequency of modulation or periodicity that they are capable to amplifying [Frisina90; Kim90]. Therefore, based on the available physiological evidence, pitch detection in the auditory brainstem is best modeled using this approach.

2.5.1 Four Stage Pitch System

To investigate the feasibility of modelling the auditory subsystem in digital VLSI, there is need to attempt the implementation of a representative system. After discussions with Meddis, the four-stage pitch system (FSPS) was chosen as the basis for the implementation work. The reasons are:

- This system is presently the best supported model, based on physiological studies, that describes pitch detection in the mammalian auditory system.
- Since a large number of cells are needed to construct a complete working system, software simulations will be very slow. Here, a hardware implementation can provide a way to realise the system in real time.
- Since this system model starts from the auditory periphery through to the brainstem, it provides a broader range of components typical in such type of systems for investigation without having to deal with the entire scope of cells that exist in the brainstem.
- The clear, feed-forward and multistage data-flow structure of this system will ease the implementation of the system. This is because each component and stage can
be designed and tested independently and the whole system can be built up in stages.

- Since this model has been simulated in software, the software model can provide a good guide to the way the system can be implemented in digital hardware. The software simulation results can be used to assess the performance of the hardware.

The basis for the four-stage model [Hewitt94] and the proposed system comes from evidence found in different parts of the auditory brainstem. Research has uncovered neural structures in the inferior colliculus (IC) that are organised according to their response to the modulation frequency, or periodicity, of the input stimulus [Langner92]. Many of these cells receive most of their inputs from cells in the cochlear nucleus which was found also to be capable of phase locking onto the modulation waveform of the stimulus [Frisina90; Kim90; Hewitt92]. These cochlear nucleus cells are found to be connected to the inner hair cells on the basilar membrane. Since all complex and natural sounds have in common their repetition rate of amplitude fluctuation, and amplitude modulation can also be perceived as pitch, Hewitt and Meddis suggested that the periodicity of the stimulus sound, represented as periodicity of the amplitude modulation in the cochlear nucleus is used by the auditory brainstem to detect pitch. The structure of their model is shown in Figure 2-8.

![Figure 2-8: The Four Stage Model for pitch detection.](image)

The cochlear mechanical filtering forms the first stage of the system. This models the function of the cochlea which separates the input stimulus into frequency bands. This
stage can be modelled by a bank of bandpass filters with different centre frequencies covering the required frequency range of input stimulus. Any sound, especially for natural sounds like speech, after passing through each filter, will produce an amplitude modulated output with a carrier frequency the same as the filter’s centre frequency. The modulation waveform of the filter’s output will depend on the type of stimulus and the characteristic of the filter.

The hair cells units form the next stage of the system. These model the inner hair cells found on the basilar membrane. Each cell converts the sound intensity and periodic information from its cochlear filter into neural spikes on the auditory nerves (AN). These spikes are generated by a transmitter substance which are produced during the positive cycle of the hair cell's input signal. The amount of transmitter substance produced is related to the intensity of the input signal as well as the saturation and adaptation characteristics of the cell. Adaptation is a process whereby production of transmitter substance increases or decreases during a sharp increase or decrease in signal intensity which is then followed by a decay over time to a steady state average rate.

The third stage is made up of an array of stellate cells. These model the cells that are found in large number in the cochlear nucleus [Frisina90]. The spikes from the hair cell travel along the AN to the dendrite of the stellate chopper cells. For a filtered input stimulus of constant amplitude, the cell will produce a neural pulse train at a relatively constant rate. This is why they are called chopper cells. From physiological studies [Frisina90] and software modelling [Hewitt92], it is known that these cells are able to detect amplitude modulations. This is because of the ability of these cells to phase lock to the modulation rate when the rate of modulation is close to the firing rate of the stellate cell. The rate where this occurs is called the best modulation frequency (BMF) of the cell. Most natural sounds, especially speech, can be broken down into a number of modulated pure tones with a modulation rate equal to the base pitch of the original sound. Therefore, the ability of stellate chopper cells to phase lock to the modulation rate will be critical to the system operation.
Each hair cell output will, therefore, feed multiple banks of chopper cells. Each bank has a different best modulating frequency (BMF) to detect a range of pitch frequencies. When the modulation rate of the signal entering a bank of stellate cells is close to the bank’s BMF, then the stellate cells will tend to phase lock to the stimulus, causing the stellate cells to spike approximately at the same time. This can be referred to as the outputs having high coincidence.

The last layer is made up of the coincidence cells. These model the cells found in the inferior colliculus which are responsible for finally converting the periodic spike information into a rate information. Each receives neural spikes from a bank of stellate cells. When a large number of stellate chopper cells in a bank spike at the same time, the coincidence cell responds by generating a spike. Hence, if the input has a pitch or modulation rate near one of the BMFs, then all the coincidence cells that are connected to banks having that BMF will have high output spike activity. This can be used to determine the pitch of the input sound.

This system for pitch detection has been modelled in software [Hewitt94] and has been partially implemented in hardware by Van Schaik [Schaik97]. But Van Schaik has only implemented a small section of the complete system which includes a cochlea and a bank of 60 stellate cells in analogue VLSI. His implementation does not include circuits to emulate the function of the auditory nerve fibres or the coincidence cell.

2.6 Summary

This chapter has briefly reviewed the physiological background needed for the understanding this thesis. Related works have also been reviewed, showing that very few implementations of the auditory system exist in digital VLSI. This chapter has also presented the suitability of the FSPS for this implementation work and has also reviewed its theory of operation.
Chapter Three
Research Overview

3.1 Objectives of Chapter

The objective of this chapter is to give an overview of the investigations in this thesis. This includes:

- Providing an overview of the research topic for the selection of investigations
- A statement of objectives
- A brief introduction to the proposed investigations, with the issues to be addressed in each investigation.
- An overview of the design flow used for the implementation work.

3.2 Selection of Research Topic

In Chapter 1, we discussed the benefits that can be gained by modelling sensory systems in hardware. We also discussed the advantage of modelling auditory systems in digital VLSI implementations, showing the potential benefit that can be gained over that of analogue VLSI implementations. However, in Chapter 2, it is clear that such a type of implementation is very much lacking, with most implementations in this area still using analogue VLSI. In the industry, there is an increasing use of digital techniques to replace tasks that were originally done in analogue VLSI. Examples can be seen in areas like control, signal processing and communications, where the digital approach results in increased flexibility, improved functionality, reduction of bandwidth or resources used, and cost savings. Therefore there is a need to determine
whether implementing in digital VLSI can indeed provide a way of modelling the auditory sub-systems effectively.

Early in 1994, a collaborative effort was started between the Electronic Systems Design Laboratory and the Hearing Research Laboratory in Loughborough University. This collaboration aims to address the use of digital VLSI implementation as a means of auditory system modelling. As a start, the FSPS was suggested by Meddis and later chosen as the basis of the implementation work in this collaboration. The suitability of this model for implementation was discussed in Section 2.5 of Chapter 2. The target system needs to operate in real-time with enough functionality to model the essential characteristics of the system, namely the pitch detection capability. Such a system will be able to help provide valuable insights into the operation of the brainstem by allowing researchers to test and measure the model using real stimuli and in real-time. It will also allow the models to be refined faster and help pinpoint areas where physiological studies need to concentrate. These insights may also, in the future, be applicable to real-world problems such as voice pre-processing for speech recognition and advanced hearing implants.

Even though the research in this collaboration is presently limited to a specific model of the auditory sub-system, knowledge gained in the implementation work can also be translated into the hardware modelling of other auditory sub-systems like sound source localisation. It may also be possible to use this knowledge to model other sensory systems like vision or olfaction, since the underlying principle of modelling biological neurons in hardware is similar.

### 3.3 Objectives of Research

Since digital VLSI implementations of the auditory system are almost non-existent, there is very little previous work on which to base our own. Hence, there is a need to address the issue of feasibility. Is it possible to implement such types of real-time systems within the limits of present and up-coming technology and to make them
reproduce the correct functionality? To tackle this issue, the following objectives are set out for the work in this thesis:

- For the system to be feasible, models have to be found for each component of the system that can reproduce the necessary behaviour while keeping the implementation size down. This means that the models have to be simple, but sufficiently accurate for emulating the essential behaviour of their biological counterparts.

- In digital systems, discretisation is unavoidable. The resolution chosen for the system will heavily affect the implementation size. This is especially so for components that exist in large numbers in the system. Therefore, the resolution and accuracy requirements of the model have to be investigated. A resolution must be chosen for the system that balances the implementation size and complexity with the functional requirements.

- The FSPS is made up of a large number of function units which in this thesis is called components. Therefore the architecture of the system has to be investigated with the objective of reducing the number of physical components and communication lines used.

- The size of the implementation has to be investigated since it will determine how realistic such an implementation will be. The size of the implementation has to be within the limits of present and up-coming technology. For example, it should be small enough for it to be implemented on a silicon chip or on a small number of field programmable gate arrays (FPGA).

- Since the aim of the work is to model a biological system in digital hardware, the entire system has to be able to reproduce the required behaviour set out in the original system model. Therefore, the behaviour of the implemented system has to be determined.

Since this implementation work is pioneering, it is essential that experience gained in this work can be applied to other similar systems. Therefore, the last two objectives, which are in many ways an overlap of the previous objectives, are as follows:
• Implementation issues that are critical to this type of system have to be identified.
• A way of dealing with the issues identified has to be proposed.

3.4 Introduction to Research Topics

3.4.1 Study of Four Stage System Component Implementation

The first step in this work is to look into the implementation of the components in the FSPS. This will be discussed in the next two chapters before dealing with the implementation of the system as a whole. For each component, there will be a need to address these issues.

• The essential functional requirements of the components need to be identified, to avoid implementing functional characteristics of the cell that are not involved in the pitch detection process.
• A computational model has to be found to meet the functional requirements for each component. These models must be simple so as to minimise the implementation size.
• A method of implementing each of these models has to be proposed.
• The implementation size and speed have to be determined.
• Finally, a comparison has to be done with biology in terms of speed and behaviour.

In Chapter 5, the stellate cell implementation is discussed separately. This is done because the stellate cells form the bulk of the system. Therefore it is critical to minimise its gate-count when attempting to reduce the implementation size of the final system. The correct operation of the stellate cell implementation is also critical to the system's functionality since it is the phase locking characteristics of these cells that contributes mainly to the detection of pitch. In the chapter, the issue of resolution and accuracy will also be investigated.
3.4.2 Study of System Implementation

After the implementation of the individual components in the system, Chapter 6 then investigates the implementation of the FSPS. In this chapter, there will be a need to address the following issues.

- Again, there is a need to determine the essential behavioural characteristics of the system as a whole that have to be captured in the implementation.
- The structuring of the system has be investigated to reduce the number of components and communication lines needed while making the system scaleable.
- The final system size and speed have to be estimated.
- Finally we will discuss whether the size and speed of the system will make implementation on a single silicon chip or a small array of FPGAs feasible.

3.4.3 System Simulation and Analysis

After choosing an architecture and the approach for implementing the system, a representative section of the system is then simulated in VHDL and Labview. In Chapter 7, we show the simulated results and analyse them to:

- Determine if the system is able to reproduce the required behavioural characteristics of the system.
- Determine what are the behaviours the system fails to reproduce and why.
- Proposed future improvements to make to system perform better.

3.5 Tools and Design Flow

Before the start of this project, the design flow was first standardised. This avoids re-engineering the test tools and set-up for each design, helping to save time and reduce errors in the test system used. Standardisation will help in the design of a very large
modular system like the FSPS by allowing each component and subsequent hierarchical level of the system to be tested using the same tightly controlled design flow, thus increasing the confidence of the modules working with each other.

![Figure 3-1: Standardised design flow.](image)

Figure 3-2: Hardware set-up

Figure 3-1 show the simplified diagram of the design flow. In it, the circuit is first described in VHDL which can then be simulated behaviourally. When the behaviour of the circuit is correct, it is then synthesised into a circuit netlist using the same stimulus file as that in the behavioural simulations, the netlist is then simulated. The response of the circuit should match that of the behavioural simulation for the netlist to be deemed correct.
To verify that the actual hardware response will match those of the VHDL and netlist simulation, the netlist is compiled and placed into an FPGA. The same stimulus file as the behavioural simulation with the expected outputs is used for testing the chip automatically. The test will give a pass or fail result indicating if the design is working correctly or not. Figure 3-2 show the hardware set-up which includes the board with the FPGA and a PC. The PC runs LabView based software that applies and reads test vectors to and from the chip using an interface card. Figure 3-3 shows the close up of the test board with the Xilinx 4025 FPGA. The Xilinx Xchecker cable for programming the chip can be clearly seen on the left of the picture. The ribbon cables connect the chip to the interface card in the PC.

Figure 3-3: Close-up of the test board with the Xilinx XC4025PG223-6 FPGA.
Chapter Four

Modelling the Components of the Four Stage Pitch System

4.1 Objectives of Chapter

This chapter looks into how the components of the FSPS can be modelled in digital hardware. The objectives are as follows:

- Give an overview of the selection of a model for each component that can be efficiently implemented in hardware and yet reproduce the necessary behaviour.
- Show how the components are implemented detailing their resultant size, speed of operation and behaviour in comparison to their biological counterparts.

4.2 Basic Four Stage Pitch System Specification

Before embarking on the implementation of components, some basic specifications for the FSPS were laid down after discussions with Meddis. These specifications, which will influence the design of the components, are as follows:

- The system must accept inputs covering the frequency range of 80.0 Hz to 5.0 kHz. This is because most of the speech energy lies in this bandwidth. Therefore, the sampling frequency of the system has to be set at least two times the nyquist rate. We choose 20.0 kHz to allow a better discrete representation of the input stimulus.
- The system should be able to detect 20 distinct pitch frequency, covering the frequency range of 80.0 Hz to 300.0 Hz. This frequency range covers the typical
The range of human pitch frequencies. Only 20 are chosen so as to provide sufficient resolution for proving the system's functionality, while limiting its size.

- The system must operate in real-time.

Along with these specifications, the general system implementation approach also has to be standardised. Other than the design flow standardisation mentioned in Chapter 3, the type of communication buses and the arithmetic techniques used will also need standardisation. These standardisations are:

- The system will use bit-serial communication between components.
- Data on the bit-serial wire will either be unsigned or in two's complement representation.
- These are presented with the least significant bit (LSB) first, with the sign bit as the last bit.
- Separate signal wires will be used to carry the clock signal and synchronisation information consisting of a pulse one clock cycle in length denoting the start of each data frame.
- All arithmetic units in the system will be implemented in bit-serial where possible.

The bit-serial technique was chosen as the basis of communication and arithmetic processes because:

- The bit-serial approach has the potential of minimising system size and the number of signal lines used in the system.
- This approach often results in very high operating speed.
- Bit-serial implementations are usually inherently pipelined. Therefore, even though this approach often results in higher latency, pipelining will compensate by increasing throughput.
- In feedforward systems, like the FSPS, higher latency will not usually affect their functionality.
Chapter 4    Modelling the Components of the Four Stage Pitch System

As a result of these advantages, bit-serial techniques are often found in the design of applications specific high speed digital signal processing systems were implementation resources are limited [For example: Summerfield92, Denyer85, Baldwin78, Lyon76].

4.3 Cochlear Mechanical Filtering

The first stage of the FSPS, as shown in Figure 2-8, represents the cochlear mechanical filtering of the auditory periphery.

4.3.1 Selection of Model

Many cochlear models exist and they are generally classified into two types. The first is based on the transmission line model. An example is that of Zweig, Lipes and Pierce [Zweig76]. This type of model attempts to capture the characteristics of the travelling wave on the basilar membrane, which is also popular among analogue VLSI implementations. The other type of models is based on an array of independent filter banks. An example is the gammatone filter [Cooke93, Chapter2] which has become predominant in physiological circles since it was able to provide a very close fit to physiological data.

However, in software modelling of large auditory systems, cochlear models tend to be based on an array of simple filters. This is because accurate filters tend to require more computational resources and therefore modellers sometimes resort to using simple filters when dealing with large systems to reduce the computational load. An example of such a compromise is shown in the work of Hewitt and Meddis where a bank of bandpass filters was used. A similar approach will be used in this digital implementation work. In this work, it is the multiple bandpass like characteristics of the cochlea that have to be modelled. So, the 2nd order bandpass filters chosen should be able to capture the essential functionality of separating the input stimulus into separate frequency bands while keeping the computational requirement low.
4.3.2 Implementation

The cochlea is implemented as a bank of 30 standard digital infinite impulse response (IIR) 2\textsuperscript{nd} order Butterworth bandpass filters shown in Figure 4-1. These filters have centre frequencies set at logarithmic frequency intervals covering the range from 80 Hz to 5 kHz, where most of the speech energy resides. Each filter has a bandwidth of 10% of its centre frequency. The filter cut-off frequency is set by changing the tap parameters at the multipliers. Only the parameters of each filter are different in the cochlear filter bank, not the design of the filter.

![Data flow diagram of a second order Butterworth IIR bandpass filter.](image)

Using 16 bit bit-serial arithmetic, Figure 4-2 shows the implementation architecture of a single cochlea bandpass filter. In the diagram, the arithmetic modules form the arithmetic engine of the filter. The parameters register is used to store the parameter values for the multipliers while the data register is used to store previous input and output values.

This filter was implemented on a Xilinx 4013PG233-6 FPGA resulting in an implementation size of 1839 gates, operating at a clock rate of 18.2 MHz. Since each epoch cycle requires 32 clock cycles, the resulting epoch rate is 568.8 kHz or the epoch cycle time is 1.7\mu s.
4.3.3 Results and Analysis

From the speed of the implementation, the iteration cycle time is much faster than the sample period. Therefore, this unit was able to operate very much faster than real-time, with the capability of emulating many units in real-time through multiplexing. Multiplexing can simply be achieved by adding the necessary data registers for all filters multiplexed by the single unit.

Figure 4-3: Response of the cochlear bandpass filter with a centre frequency of 3350Hz to random noise.
Figure 4-3 shows that the response of a filter with a centre frequency of 3350 Hz to a random noise input resembles a randomly modulated sinusoidal waveform. The sinusoidal output has the same carrier frequency as the centre frequency of the bandpass filter (BPF) itself. This shows that the filter is operating as expected, extracting an amplitude modulated signal from the stimulus.

4.4 Mechanical to Neural Transduction

The second layer of the FSPS represent the cochlea mechanical to neural transduction performed by the inner hair cell.

4.4.1 Selection of Model

The inner hair cell has been modelled in various ways by many researchers. These range from simple spike generators, like those by Lazzaro and Mead [Lazzaro89a], to simple saturation and adaptation models like Van Schaik's hair cell implementation [Schaik96b], to those that attempt to model the biochemical process involved, like the hair cell model by Meddis [Meddis86]. For the implementation of the FSPS, the selected model has to be able to simulate the half-wave rectification, saturation and adaptation characteristics of the hair cell. These are necessary because the saturation and adaptation characteristics are involved in compressing the dynamic range of the input stimulus through their non-linear characteristics, while half-wave rectification is essential for extraction of the modulating waveform.

In selecting the appropriate model for this implementation, we only considered models that are computational. Through Hewitt and Meddis's review [Hewitt90] and discussion with Meddis, the hair cell model of Meddis [Meddis86] was chosen for the following reasons:
Chapter 4 Modelling the Components of the Four Stage Pitch System

- It is able to simulate all the required characteristics of the hair cell, which include both the short term and long term adaptation characteristics of the hair cell.
- The model is computationally efficient, using only three simple differential equations and one saturation function.
- This was also the model used by the Hewitt and Meddis in their software simulation work on the FSPS.

4.4.2 The Meddis Inner Hair Cell Model

The Meddis inner hair cell model uses a set of four equations to model the behaviour of the inner hair cell. These equations are listed in Figure 4-4 and they model the process that ejects the transmitter into the synaptic cleft when stimulated. The first equation models the rectification and saturation characteristics of the hair cell while the other three models the flow of transmitter substance between the synaptic cleft, the reprocessing pool and the free transmitter pool. These are represented diagramatically in Figure 4-5.

\[
\begin{align*}
(1) \quad k(t) &= \begin{cases} 
  \frac{gdt \cdot [s(t) + A]}{s(t) + A + B} & \text{for } [s(t) + A] > 0 \\
  k(t) = 0 & \text{for } [s(t) + A] \leq 0 
\end{cases} \\
(2) \quad \frac{dw}{dt} &= r.c(t) - x.w(t) \\
(3) \quad \frac{dq}{dt} &= y.(1-q(t)) + x.w(t) - k(t).q(t) \\
(4) \quad \frac{dc}{dt} &= k(t).q(t) - l.c(t) - r.c(t)
\end{align*}
\]

\[k(t)\] permeability
\[w(t)\] reprocessing store
\[q(t)\] free transmitter pool
\[c(t)\] cleft content
\[s(t)\] stimulus
\[A, B\] for setting saturation function and spontaneous level.
\[g\] transmitter release gain
\[r\] rate of return from cleft
\[x\] rate of return to free transmitter pool
\[y\] rate of replenishment
\[l\] rate of loss from cleft

Figure 4-4: Meddis inner hair cell model equations.

The reprocessing pool simulates the slower process of the recycling of some of the transmitter substances already in the synaptic cleft. The free transmitter pool models the store of transmitter substance ready to be ejected to the synaptic cleft while the
cleft content models the level of transmitter substance in the synaptic cleft, which is dependent on the rate of ejection, the reprocessing rate and the rate of loss. These, interacting together, are able to reproduce the various adaptation characteristics of the system because the contents in the free transmitter pool, containing the transmitters ready to be ejected into the synaptic cleft, will vary according to the intensity of the input stimulus and the rate of replenishment. The output of the hair cell is a variable that describes the level of transmitter substances in the synaptic cleft which will then be used as a probability value to vary the amount of spiking activity in the auditory nerve fibres.

![Flow diagram of the Meddis inner hair cell model](image)

Figure 4-5: Flow diagram of the Meddis inner hair cell model.

### 4.4.3 Implementation

The Meddis inner hair cell model was implemented by A. R. Temple using 16 bit bit-serial arithmetic. The implementation is made up of three top level units as shown in Figure 4-6. The CALC_FLOW unit calculates all the required flow of transmitters per epoch while the CALC_RES unit updates the three variables $q$, $c$ and $w$ according to the calculated flow results from the CALC_FLOW unit. The REG_SET unit is used to latch and/or store all the variables in the system including the input bit stream. All constants in the design are hardwired which helps to save on implementation size. This is done because there is only a need for a single type of hair cell in the FSPS. Parameters for modelling a high spontaneous rate hair cell are extracted from Meddis's papers [Meddis86; Meddis88; Meddis90] for use in the implementation. The high spontaneous rate fibre is chosen because it provides the maximum and most appropriate amount of compression needed for the system. The unit was implemented
on a Xilinx XC4013 FPGA device. A maximum clock rate frequency of 22.2 MHz was achieved. A new result was produced every 368 clock cycles, resulting in an epoch cycle rate of 60.3 kHz or a time of 1.7 ms. The implementation size is 2789 gates.

Figure 4-6: Data flow structure of the inner hair cell implementation by Temple.

4.4.4 Results and Analysis

Figure 4-7: Response of the Meddis hair cell implementation to step intensity sinusoid. (A) Input Stimulus, (B) Hair cell implementation's results, (C) Floating point software implementation results from Meddis [Meddis86, Fig11]
From the speed of implementation, we can deduce that the unit will be able to operate in real-time. To test the implementation, the unit was stimulated with a 1 kHz pure tone stimulus as shown in Figure 4-7(A), starting with 0 dB, stepping to 50 dB and finally 70 dB. The response of the 16-bit model in Figure 4-7(C) clearly shows the onset characteristics, when the stimulus change its intensity, and the adaptation characteristics of the hair cell. This performance is very close to that of the hair cell simulations by Meddis [Meddis86], as shown in Figure 4-7(B), which use floating point arithmetic.

Since the hair cell unit is to be used in the FSPS, it is essential that the hair cell implementation is able to preserve the modulation information of any modulated stimulus while performing half-wave rectification. To test this, the hair cell implementation was stimulated by a 30 dB intensity, 25% modulated sinusoid. The results in Figure 4-8 show that the modulation information was preserved, though with some loss of modulation due to the saturation characteristics of the hair cell. The output is also half-wave rectified as expected.

Figure 4-8: Response of the hardware Meddis hair cell model to a 30 dB, 30% modulated sinusoid.
Chapter 4  Modelling the Components of the Four Stage Pitch System

4.5 Auditory Nerve Fibre

The auditory nerve fibres connect the inner hair cells to the cochlear nucleus. These nerve fibres carry the sound stimulus as nerve impulses. Even though they are not classified as a separate stage in the FSPS, they are essential for the system to work correctly.

4.5.1 Selection of Model

Each hair cell in the auditory system will have many auditory nerves connected to it. In the FSPS, these nerves will then connect to the stellate cells. Each of these auditory nerve fibres exhibits a Type I or primary-like [Pickles88, Chapter 6] characteristic, meaning that they fire randomly when stimulated by neurotransmitters, with the rate of firing dependent on the amount of neurotransmitter substance present. The amount of neurotransmitters can be represented as the spike probability for each nerve fibre. A large number (typically around forty for each stellate cell in the system) of fibre is required in the system for two reasons. The first is that the noise inherent in the fibres will help to desynchronise the stellate cell spiking response except when the best modulation frequency of the stellate cell matches that of the modulation frequency. The second reason is that the auditory nerve fibres can only fire up to a rate of 1 kHz, therefore requiring many fibres to represent the higher frequency components in the input stimulus.

![Diagram of auditory nerve fibre interconnections between hair cells and stellate cells.](image)

Figure 4-9: Diagram of auditory nerve fibre interconnections between hair cells and stellate cells.
The auditory nerve fibres are usually modelled as simple spike generators in hardware. Examples of these simple implementations are those of Lazzaro and Mead as used in their localisation and pitch extraction circuits [Lazzaro89a; Lazzaro89b]. Other more advanced hardware models like that of Lazzaro [Lazzaro92a] include simple adaptation behaviour that we choose to model in our system using the hair cell implementation. None of the hardware models found in our review attempt to model the random spiking nature of the auditory nerves except through the random inaccuracy of the implementation themselves. In software modelling, many types of methods exist for the modelling of the auditory nerve fibres. A simple but common method uses a random number generator with the probability value as a threshold to determine if a spike will be generated. Additionally, a timing algorithm is used to simulate the refractory period. An example of this is that used by Meddis [Meddis90]. Another example, but further extended, is that of Carney [Carney93].

In the FSPS, since each stellate cell in the system would require a set of fibres to be connected to them, as shown in Figure 4-9. Using any of these types of models would result in an unacceptable number of these units in the entire system. Therefore, a simpler method has to be found.

![Diagram of simplified method of simulating nerve fibres.](image)

Since a large number of nerve fibres connect each hair cell to each stellate cell, and the input of the stellate cell consist of a dendrite that lowpass filters the signal, it is reasonable to suggest the output of the stellate cell dendrite will be a noisy lowpass filtered version of the hair cell probability value. Based on this idea, Figure 4-10
shows a method of overcoming this problem. This model approximates the system by simply adding a scaled noise to the probability value from the hair cell. A function for the scaling of the noise, proposed by A.R.Temple, is shown in Figure 4-11. This function will provide no noise when the probability is zero or one, because there will be no noise when no fibre spikes or when they all spikes continuously. Since \( p \) is usually very small, it is therefore possible to simplify and approximate the equation shown in Figure 4-11 to just that of \( S(p) = G \cdot p \).

\[
S(p) = G \cdot (1 - p) \cdot p
\]

\( G \) Noise gain

\( p \) Probability (Output from hair cell)

**Figure 4-11:** Noise scaling function, \( S(p) \).

### 4.5.2 Implementation

To implement the noise source, a simple linear feedback shift register (LFSR) [Tsui86, Page172-180] is used. This unit is used to supply a random serial bit stream. The arithmetic units are implemented using 16 bit bit-serial arithmetic. Figure 4-12 shows the architecture of the auditory nerve fibre unit. Note the use of the AND gate in the circuit to perform the scaling of the noise source. This is loosely based on the stochastic arithmetic system by Mars and Poppelbaum [Mars81]. The manipulate sign unit functions as a negator so that when the most significant bit of the noise source...
is '1', the resultant scaled noise source is negated. This allows the noise not only to be added to the stimulus, but also to be subtracted from it. This unit was implemented on a Xilinx 4013PG223-6 FPGA resulting in an implementation size of 808 gates and a clock speed of 32.1 MHz. Each epoch requires 32 clock cycles resulting in an epoch rate of 1.0 MHz or an epoch time period of 1 μs.

4.5.3 Results and Analysis

The implementation speed of this unit makes it possible for it to emulate up to 50 units through multiplexing. To test it, the output of the inner hair cell implementation shown in Figure 4-8 was applied to the unit. Figure 4-13 shows its output which is a noise added version of the hair cell output, as expected.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure4-13.png}
\caption{Result of the auditory nerve fibres unit when stimulated by the output of the inner hair cell unit. The hair cell output probability is represented as a voltage output in this diagram.}
\end{figure}

4.6 Coincidence Cell

The last stage of the system is made up of cells that detect the phasing locking of the stellate cell's output spikes by determining if more than a set number of spikes occur at any one time.
4.6.1 Selection of Model

Since the only essential characteristic of the unit is to detect more than a number of spikes at any one time, to model this unit, we chose to use a simple comparator system to determine if the cell should spike. The input spikes, as shown in Figure 4-14, are first summed before comparing it to a pre-set threshold value. If the sum of the input spikes exceeds the threshold value, the cell will then produce a spike.

![Data flow diagram of the coincidence cell model.](image)

**Figure 4-14:** Data flow diagram of the coincidence cell model.

4.6.2 Implementation

![Simplified architecture of the coincidence cell unit.](image)

**Figure 4-15:** Simplified architecture of the coincidence cell unit.

Figure 4-15 shows the architecture of the coincidence cell implementation. This implementation is made up of two main units called the spike counter unit and the Comparator Unit. The spikes from the stellate cells are supplied serially into the Spike
Counter Unit, which is then counted by the adder and shift register setup. The counter then outputs the result, after a fixed number of epoch cycles representing a bank of stellate cells, to the Comparator Unit. The Comparator Unit then compares the input value to the threshold by subtracting them. It then outputs a binary value which determines if a spike is generated in this epoch cycle. The AND gates in the circuit are used to format the data streams when changing from faster to a slower data rates. When implemented onto a Xilinx XC4013PG233-6 FPGA, the size of the unit is 111 gates operating at a maximum clock rate of 32.1 MHz. With each epoch cycle taking 32 clock cycles, the epoch rate is 1.0 MHz or the epoch cycle time is 1 μs.

4.6.3 Results and Analysis

Based on the implementation speed, it is again possible to multiplex a single unit in real-time to emulate up to 50 units. To test it, we supplied the unit with the spiking output of a bank of stellate cells. The results in Figure 4-16 show that whenever the sum of the spikes from the stellate cells exceeds the threshold value, an output spike is produced.

![Figure 4-16: Input and output waveforms of the coincidence cell.](image)

<table>
<thead>
<tr>
<th>Total spikes per epoch from a bank of stellate cells</th>
<th>Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

| Coincidence cell output | 0 | 1 | Time (ms) | 0 | 100 |
4.7 Summary

This chapter has shown how most of the components of the FSPS can be implemented. These important characteristics needed for each component have been identified and the appropriate models were selected to reproduce them. A summary of the speed and size of the components discussed in this chapter is shown in Table 4-1. These figures show that it is possible to achieve real-time operation with multiplexing to reduce the component count in the system. The implementation size of each component is also small, making it possible to implement many of these units on hardware to form large systems. These are in line with the objectives laid out in the beginning of the chapter.

<table>
<thead>
<tr>
<th>Component Name</th>
<th>Epoch Rate</th>
<th>Implementation Gate Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cochlear Bandpass Unit</td>
<td>568.8KHz</td>
<td>1839</td>
</tr>
<tr>
<td>Hair Cell Unit</td>
<td>60.3KHz</td>
<td>2789</td>
</tr>
<tr>
<td>Auditory Nerve Fibre Unit</td>
<td>1MHz</td>
<td>808</td>
</tr>
<tr>
<td>Coincidence Cell Unit</td>
<td>1MHz</td>
<td>111</td>
</tr>
</tbody>
</table>

Table 4-1: Summary of speed and size of components in this chapter.
Chapter Five

Implementation of the Stellate Cell

5.1 Objective of Chapter

This Chapter looks into the implementation and experimentation of the stellate cell. The objectives are to:

- give an overview on the selection of a model for the stellate cell that can be efficiently implemented in hardware and yet reproduce the necessary behaviour
- show how it is implemented and determine its resultant implementation size, speed of operation and behaviour in comparison to its biological counterparts
- investigate the impact of varying the resolution of the implementation on accuracy.

5.2 Selection of Model

In the FSPS, the third stage is made up of stellate cells. These represent cells that are found in the cochlear nucleus of the auditory brainstem. They are also called chopper cells because, when they are stimulated by a pure tone signal, they produce a consistent and constant train of spikes that is independent of the stimulus frequency. Physiological studies have revealed that these cells are able to use their spiking capability to phase lock onto the modulation signal of a amplitude modulated stimulus when their spiking rate matches the modulation rate. It is this characteristic that the FSPS depends heavily for the detection of pitch. Therefore, the selected model must be able to reproduce this behaviour.
There are many ways of modelling spiking neurons. These ranges from the simple integrate and fire approach, like that of Lazzaro [Lazzaro92b], to complex models that describes the cellular process in the generation of these spikes, for example Hodgkin and Huxley [Hodgkin52c]. However, in order to model the phasing locking characteristics of the stellate cell, the latter approach has to be used because simple integrate and fire approach are unable to capture this characteristic.

The first such model of the cellular process of spike generation was formulated by Hodgkin and Huxley [Hodgkin52a; Hodgkin52b; Hodgkin52c] in 1952. They used a set of equations to describe how the membrane potential of a neuron is affected by various resting and active membrane conductances (rate of ionic flow). However, these equations are not able to model sufficiently the continuous spiking process since they are formulated to represent the mechanism behind a single-action potential formation [MacGregor87, Chapter 2]. Much work has since been done to allow continuous processes to be modelled. This includes modifications to the Hodgkin and Huxley models. However, for simulations of larger systems on a computer, these models are usually simplified. An example is the work of MacGregor [MacGregor87] which uses a set of four simple differential equations. Other examples are those by Arle and Kim [Arle90; Arle91], which uses an extended version of MacGregor's model, and the FitzHugh-Nagumo model [FitzHugn61; Barranco91], which uses a time-dependent non-linear function to model the sodium current fluctuations.

For the stellate cell in the FSPS, the modelling method chosen is based on that formulated by MacGregor [MacGregor87, Chapter 14] which is a simplified approximation of the cellular process that is computationally less intense and therefore possibly simpler to implement in hardware. The modelling method used by MacGregor compartmentalises the typical neural cell into different parts consisting of synaptic inputs, dendrite model, soma model, and output.
dendrite trees and a cell soma as shown in Figure 5-1. Following the method used by Hewitt [Hewitt92], we chose to base the implementation of the cell soma on MacGregor's State-Variable Point Model for Repetitive Firing of Neurons (PTNRN10) [Macgregor87, Chapter 14, Part A] since it is computationally efficient. Similarly, the dendrite implementation will be based on a simple low pass filter.

5.3 Dendrite

The dendrite model implements the low pass filtering characteristics of the dendrite of the stellate cell in the FSPS.

5.3.1 Low Pass Filter Model

Since the input to a stellate cell from the hair cell is simplified from many auditory nerve fibres to just a noisy signal, the dendrite model can be dramatically simplified. This can be done by using a single lowpass filter to model the lowpass filtering characteristics required for the stellate cell. Based on Hewitt's experimental results [Hewitt92], the 1st order butterworth lowpass filter with a cut-off frequency of 300 Hz is selected.

5.3.2 Implementation

The dendrite filter is implemented as a standard digital IIR 1st order butterworth lowpass filter as shown in Figure 5-2.

![Data flow diagram of a 1st order Butterworth IIR lowpass filter.](image)
Using 16 bit bit-serial arithmetic, Figure 5-3 shows the implementation architecture of a single cochlear bandpass filter. In the diagram, the arithmetic modules form the arithmetic engine of the filter. The parameters are hardwired because there is only one type of lowpass filter in the system. The data registers are used to store previous input and output values.

Figure 5-3: Architecture of a dendrite lowpass filter. The clock signal, which is not included in this diagram, connects all modules except for the AND gates.

This filter was implemented into a Xilinx 4013PG233-6 FPGA resulting in an implementation size of 634 gates. The implementation was able to operate at a clock rate of 39 MHz. Since each epoch cycle requires 32 clock cycles, the resulting epoch rate is 1.21 MHz or the epoch cycle time is $0.82\mu s$.

### 5.3.3 Results and Analysis

The above figures indicate that the iteration cycle time is much faster than the sample period. Therefore, this unit is able to operate very much faster than real-time, with the capability of emulating many units in real-time through multiplexing. Multiplexing can again be achieved by adding the necessary data registers for all filters to be emulated by the single unit. To test this unit’s functionality, the output of the auditory nerve unit is used as the input into this unit. Figure 5-4 shows the lowpass filtering of
the noisy halfwave rectified signal revealing the modulation signal with a dc offset as expected.

![Graph](image)

Figure 5-4: Response of dendrite lowpass filter to the output from an auditory nerve fibre unit.

### 5.4 Stellate Cell Soma

The stellate cell soma models the non-linear, spiking characteristics of the cell which will phase lock to the modulation frequency.

#### 5.4.1 Cell Soma Model

The cell-soma model used here for the simulation of the auditory cell is based on the PTNRN10 model [Macgregor87, Chapter 14, Part A]. This model is able to produce the approximate input/output dynamic characteristics of a point neuron without dendrites. It is based on four first-order differential equations as shown in Figure 5-5 with $E$ representing the cell's transmembrane potential, $Th$, the threshold, $S$, the spiking variable, and finally $GK$, the potassium conductance of the membrane.

The first equation simulates the effect of the internal and external ion fluctuations on the transmembrane potential. $T_{mem}$ represents the membrane time constant, $Ek$, the potassium equilibrium potential and $Vin$, the input voltage. The second equation deals with the threshold potential, $Th$, with its accommodation (adaptation) ability. Here, $Th(0)$ is the resting threshold, $C$ is the threshold accommodation constant and $TTh$, the
threshold accommodation time constant. In Equation 3, $S$ is a boolean variable representing the cell spike state. When the transmembrane potential exceeds the threshold, the cell will fire which is represented by $S$ set as 1. Lastly, the equation 4 describes the potassium conductance, where $B$ is the delayed rectifier potassium conductance strength, $S$ the spiking variable and $TGk$ the potassium conductance delay time constant. Figure 5-6 shows an example of how the variables interact.

\[
\begin{align*}
\frac{dE}{dt} &= \frac{-E + V(t) + Gk(Ek - E)}{T_{mem}} \\
\frac{dTh}{dt} &= \frac{-(Th - Th(0)) + C(E)}{T_{Th}} \\
S &= \begin{cases} 
0 & E < Th \\
1 & E \geq Th 
\end{cases} \\
\frac{dGk}{dt} &= \frac{-Gk + B(S)}{TGk}
\end{align*}
\]

Figure 5-5: PTNRN10 model equations.

An additional equation, below is used for producing the approximate output spiking potential, with $Eb$ representing the reversal potential of the cell (the strength of the spike).
Chapter 5 Implementation of the Stellate Cell

\[ p(t) = E(t) - S(E_b - E(t)) \]  

Figure 5-7: Equation for approximating the output spiking potential.

The instantaneous input potential of the cell is proportional to instantaneous injected current and this relationship is given below with \( R \) representing the input resistance of the cell. This is to allow us to compare our results with biological experimental data by converting stimulating currents into voltages.

\[ V(t) = I(t)R \]  

Figure 5-8: Equation for membrane resistance.

Simulation of this model in software or digital hardware can be performed by discrete integration. For example, the threshold equation can be implemented by the equation below, using previously calculated values to calculate the next discrete time value. Simulation accuracy here depends very much on the epoch time \( \Delta t \) and will approach the ideal case when \( \Delta t \) tends to zero.

\[ Th(t) = Th(t-1) + \Delta t \left( -\frac{(Th - Th0 + C(E))}{TTh} \right) \]  

Figure 5-9: Discrete integration implementation of the threshold equation.

Since MacGregor's equations are formulated based on the epoch time of 1ms and simulated by converting these to exponential equations, the \( Gk \) equation needs to be slightly modified and implemented differently to allow for shorter epoch time simulation using the discrete integration method. Here, \( S' \), the positive going edge of \( S \), is used instead to avoid the value \( B \) being added to \( Gk \) more than once in the same spike.
Chapter 5

Implementation of the Stellate Cell

\[
G_k(t) = \Delta t \left( -\frac{G_k}{Tg_k} \right) + B(S')
\]

(8)

**Figure 5-10: Modified Gk equation.**

Since the MacGregor point neuron is formulated to model many types of neurons, the parameters of the model need to be defined to model the stellate cell soma specifically. The stellate cell fits well into this model, and an abundance of biological data [Ortel 83; Ortel85; Feng94; Kim90] exists for the estimation of the parameters and the comparison of the results. Some parameters, like the membrane time constant, input resistance and threshold were specified by Ortel [Ortel85], while others were derived empirically to approximate the biological results. The parameters used are shown in table 5-1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>0.30</td>
<td>dimensionless</td>
</tr>
<tr>
<td>TTH</td>
<td>20.00</td>
<td>ms</td>
</tr>
<tr>
<td>B</td>
<td>1.70</td>
<td>ns</td>
</tr>
<tr>
<td>TGK</td>
<td>0.90</td>
<td>ms</td>
</tr>
<tr>
<td>TH0</td>
<td>10.00</td>
<td>mV</td>
</tr>
<tr>
<td>TMEM</td>
<td>2.00</td>
<td>ms</td>
</tr>
<tr>
<td>R</td>
<td>33.00</td>
<td>MΩ</td>
</tr>
<tr>
<td>Ek</td>
<td>-60.00</td>
<td>mV</td>
</tr>
<tr>
<td>Eb</td>
<td>0.00</td>
<td>mV</td>
</tr>
</tbody>
</table>

Table 5-1: Parameter values for the stellate cell soma model.

Another advantage of modeling the stellate cell is that this cell has little adaptation. This allows the threshold equation to be ignored without adversely affecting the behavioral characteristics and therefore reducing the equation to a constant. Figure 5-11 shows a comparison between two simulations with and without using the threshold equation along with Ortel’s data. Note that the difference in spike rate between Figure 5-11(b) and Figure 5-11(c) is negligible. Eliminating the need for a varying threshold simplifies the model and narrows the scope of resolution investigation to be performed later to just two variables.

In the FSPS, cells with many different spike rates will be used. This can be done by adjusting the parameter values $T_{mem}$ and $T_Gk$. However, in order to make sure that
the threshold value can be kept constant for all these cells, these two time constant values have to be chosen such that the positive peak and negative peak of the variable \( E \) are approximately the same. Through simple experimentation, we found a rule of thumb that can be used to achieve this balance is to set the value for \( TGk \) at approximately half that of \( Tmem \).

Figure 5-11(a): Actual neural response to depolarising current of 0.6 nA [Ortel 1983, figure 5], plotted with reference to the cell's resting potential.

Figure 5-11(b): Stellate cell-soma model response plotted with reference to the cell's resting potential.
Chapter 5

Implementation of the Stellate Cell

Figure 5-11(c): Stellate cell-soma model response with threshold set to a constant, plotted with reference to the cell's resting potential.

5.4.2 Implementation

Figure 5-12 shows the simplified implemented data flow diagram of the unit. In the diagram, the $S'$ variable defines the positive going edge of the spike variable which is used to determine if the value $B$ is to be added to the $Gk$ value. The registers labelled $z^{-1}$ are delay registers, delaying the variable by one epoch time. $S'(t)$ is used as the output value so that the output of the cell will not produce consecutive spike values of 1 which will produce an erroneous result at the coincidence cells in the last stage of the FSPS.

Figure 5-12: Data flow diagram of the stellate cell soma unit.
Implemented using 16 bit bit-serial arithmetic, the resulting architecture is shown in Figure 5-13. In the diagram, the parameter registers are used for storing the two required parameters, while the data registers are used for storing the previously calculated variable values. The delay registers are used for synchronising the different data lines in the circuit. Note that in the design, a simple AND gate is used instead of the multiplexer shown in Figure 5-12 since one of the values is always zero.

Figure 5-13: Architecture of stellate cell soma implementation.

This unit was implemented on a Xilinx 4013PG233-6 FPGA with an implementation size of 1780 gates. The implementation operated at a clock rate of 32.1 MHz. Since each epoch cycle requires 32 clock cycles, the resulting epoch rate is 997 kHz or an epoch cycle time of around 1 μs.

5.4.3 Results and Analysis

Based on the epoch speed of the stellate cell soma unit, it is possible to multiplex this unit to simulate up to 49 units in real-time. To test the functionality of the unit, the
stellate cell unit is loaded with the parameters listed in Table 5-1. With the same stimulus applied, the results are shown in Figure 5-14. Figure 5-14(D) shows the approximate membrane spiking potential derived from Figure 5-15 (A) and (C) by using Equation 5. The results show good correspondence to Ortel's data shown previously in Figure 5-11(a). Since only the spikes of this unit are required as the output, the waveform as shown in Figure 5-14 (D) does not need to be generated by the implementation.

Figure 5-14: Response of stellate cell soma implementation to 20mV stimulus. Graph (D) is the approximate spiking potential derived by using equation (5).

To test if this unit is able to phase lock to the modulation waveform, the output of the dendrite filter as shown in Figure 5-4 was applied to the unit. The parameters of the unit were set to provide a best modulation frequency similar to the modulation frequency. The results are shown in Figure 5-15. In the diagram, the spike output of
the stellate cell soma implementation phase locks to the envelope's peak values as expected after the first few spikes.

![Dendrite Filter Output](image1)

**Figure 5-15:** Stellate cell soma response to the output from the dendrite filter shown in Figure 5-4.

To show how a bank of stellate cells will respond to modulated signals, 40 similar stellate cells were simulated in software. A cochlear filter unit, a hair cell unit and a bank of 40 auditory nerve fibre units were also simulated prior to this to provide the inputs to the stellate cells. Figure 5-16 shows the results of the simulation when the test system was stimulated by a 25%, 198 Hz modulated 3350 Hz pure tone. Three BMFs were used and the results shows that when the modulation frequency matches the BMF, the sum of spikes from the stellate cells has the highest peak values. This means that the bank with the same BMF as the modulation of the stimulus is showing a much higher amount of phase locking as compared to the other two, indicating that it is detecting the pitch of the stimulus.

![BMF Outputs](image2)

**Figure 5-16:** Sum of the outputs of 40 stellate cells with different BMF when stimulated by the same modulated pure tone.
5.5 Cell Soma Resolution Experimentation

5.5.1 Motivation and Experimentation Set-up

The MacGregor cell models are formulated for software simulations. Usually in software simulations, floating point numbers or high bit width number representations are used. However, when the MacGregor cell model is to be implemented into hardware circuit where resources are limited, the epoch time and the resolution used have to be reduced to reduce system size. These will in turn introduce quantisation errors and affect the accuracy and performance of the model. This is especially true in systems that use multiple interacting recurrent equations, where errors are amplified. Therefore there is a need to investigate the errors resulting from the constraint of resolution and epoch time in the stellate cell soma unit.

In the stellate cell soma unit, there are three variables that can affect the accuracy and functionality of the system. They are the epoch time period and the resolution of $E$ and $G_k$. In the implementation approach chosen previously, using smaller epoch time period will provide higher accuracy. However, using small time periods will also mean that higher resolutions are needed for the variables since the increments in $G_k$ and $E$ are smaller. Therefore to find a compromise, a set of experiments was performed, varying these three variables to determine the amount of error introduced. Error is measured by calculating the average root mean square pulse period error between a double-floating point simulation with very small epoch time steps (0.00001 ms) and the contrainted bit-width and epoch time software based simulations. The parameters listed in Table 5-1 were used for this experiment.

5.5.2 Experiment Results and Analysis

Figure 5-17 shows a representative subset of the results of the experiments with $dt$ representing the epoch time period and the number next to each variable, $G_k$ and $E$, referring to their fractional bit resolutions. The fractional resolutions of the variables are in addition to the variable whole bit resolution of 8 bits for $E$ and 3 bits for $G_k$. 
Figure 5-17(a,b,c,d,e,f): Average pulse period error in relation to Gk, E fractional bit resolution and the epoch period dt.
Chapter 5

Implementation of the Stellate Cell

It is clear that from the results in the two experiments, that resolution and epoch time are closely related. When $dt$ decreases, the accuracy of the model improves only to a point and then starts deteriorating. For example, in Figure 5-17(f), the result for the fractional resolution of 11 for Gk and 13 for E, the maximum error envelope decreases with $dt$ until around 0.08 ms. After this point, the error envelope increases dramatically. The deterioration is due to underflow when the incremental values become too small to be represented by the bit-width resolution used. In the results, the minimum error point shifts progressively to higher $dt$ values as the resolution used decreases. Therefore, to maintain accuracy at lower $dt$ values, the resolution has to be increased.

It is known that biological systems operate with significant inaccuracies, but this is compensated through averaging the results of many inaccurate cells. However, in digital implementation, where multiple identical units are used, the error generated by each unit will be the same with the same inputs, making it impossible to reduce the errors by averaging. Therefore a higher degree of accuracy is needed. Since the exact accuracy needed for the implementation is not known, the selection of accuracy will depend on the need of the system. An approach however exists to compensate for this. Inaccuracies in the choice of resolution cause a change in the pulse period. Therefore, it may be possible to compensate for small shifts by simply varying the parameters of the cell soma unit slightly.

To choose the required variable resolutions and the epoch time, the first step will usually be to determine the $dt$ time required. A simple approach will be to choose an epoch time that is the same as the sample period of the input sound of the system. After choosing the epoch time, the fractional bit resolutions for $E$ and $Gk$ are then selected based on the required accuracy. This is performed by comparing the maximum error envelopes of the pulse period error charts for different $E$ and $Gk$ fractional bit resolutions at the chosen epoch time period. Examples of such selections are shown below in Table 5-2. The selection of the resolutions will also be influenced by practicality in implementation. For example, if standard memory modules are to be used for storage of these values, it will be easier to select resolutions based on powers of two (i.e. 4, 8 or 16 bits). Matching the bit width of $E$
and $G_k$ components will also help in simplifying the design work. Two possible choices considering these issues are 13 bits, when not using powers of two, and 16 bits for both $E$ and $G_k$ bit widths. These will provide at least 90\% and 97.5\% accuracy respectively.

<table>
<thead>
<tr>
<th>Accuracy %</th>
<th>Sample rate kHz</th>
<th>$d_t$ ms</th>
<th>E fractional bit resolution</th>
<th>E total bit resolution</th>
<th>Gk fractional bit resolution</th>
<th>Gk total bit resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>20</td>
<td>0.05</td>
<td>5.00</td>
<td>13</td>
<td>8.00</td>
<td>11</td>
</tr>
<tr>
<td>90</td>
<td>20</td>
<td>0.05</td>
<td>5.00</td>
<td>13</td>
<td>9.00</td>
<td>12</td>
</tr>
<tr>
<td>97.5</td>
<td>20</td>
<td>0.05</td>
<td>8.00</td>
<td>16</td>
<td>10.00</td>
<td>13</td>
</tr>
</tbody>
</table>

Table 5-2: Resolution selection for different accuracy requirements for the stellate cell soma.

5.6 Summary

This chapter has shown how the stellate cell can be implemented as two separate components. The dendrite of the stellate cell can be implemented as a low pass filter while the cell soma implementation can be based on a simplified version of the MacGregor point neuron model. These components are able to reproduce the required functionality at many times the required real-time speed and have small implementation size.

<table>
<thead>
<tr>
<th>Component Name</th>
<th>Epoch Rate (kHz)</th>
<th>Implementation Gate Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cochlear Bandpass Unit</td>
<td>568.8</td>
<td>1839</td>
</tr>
<tr>
<td>Hair Cell Unit</td>
<td>60.3</td>
<td>2789</td>
</tr>
<tr>
<td>Auditory Nerve Fibre Unit</td>
<td>1000</td>
<td>808</td>
</tr>
<tr>
<td>Coincidence Cell Unit</td>
<td>1000</td>
<td>111</td>
</tr>
<tr>
<td>Dendrite Filter Unit</td>
<td>1200</td>
<td>634</td>
</tr>
<tr>
<td>Stellate Cell Soma</td>
<td>997</td>
<td>1780</td>
</tr>
</tbody>
</table>

Table 5-3: Summary of speed and size of all components.

A summary of the implementation size and speed of the two components in this chapter is shown in Table 5-3 along with the rest of the components in the system. Note that the largest component in the system is the hair cell unit. Fortunately, the
number of these in the system is very much less than the other components, therefore reducing its impact on system size. The issue of system size will be discussed in Chapter 6 based on these component size results.

Appropriate bit-width and epoch time resolutions were also chosen through experimentation for the stellate cell soma unit to balance implementation size, practicality and accuracy. These pave the way for the design and implementation of large arrays of these cells needed for the FSPS efficiently.
Chapter Six
Pitch System Implementation

6.1 Objectives of Chapter

This chapter looks into the implementation and simulation of the FSPS. The objectives are to:

- Give an overview on the issues involved in choosing the appropriate architecture for the FSPS.
- Investigate the use of multiplexing as a ways of reducing the number of components needed for the FSPS.
- Show how the complete system can be implemented and estimate the resultant implementation size and speed of operation.
- Show how the system was implemented in software to test the system’s functionality.

6.2 System Architecture

6.2.1 Architecture Based on a One-to-One Approach

Chapter 4 mentioned that the FSPS will need to accept inputs covering the frequency range of 80 Hz to 5 kHz and detect 20 possible pitch frequencies from 80 Hz to 300 Hz. To cover the input frequency range, 30 cochlear filter units are needed. Since each cochlear filter unit will have a hair cell connected to it, there will also be a need for 30 hair cells. To detect 20 pitch frequencies, each hair cell needs to connect to 20 banks of 40 stellate cells each through an auditory nerve fibre unit. Each stellate cell
in a bank will then connect to a coincidence cell unit which then produces an output. The output of the system will therefore be made up of an array of 40x20 coincidence spiking outputs. We call a single structure from a cochlea filter channel to its corresponding coincidence cells a *cochlea channel*. Figure 6-1(a) shows the top level architecture of the entire system while Figure 6-1(b) shows the data flow architecture of a single cochlear channel.

**Figure 6-1:** (a) Top level data flow architecture of the FSPS. (b) Architecture of a cochlear channel based on one to one approach.
This system will operate at an epoch rate of 20 kHz, which is equal to the input stimulus sample rate. Additional multipliers are included in the architecture to allow the stimulus level to be adjusted before being applied to the system and also before the stellate cells.

<table>
<thead>
<tr>
<th>Component</th>
<th>Component Count</th>
<th>Gate Size Each</th>
<th>Total Gate Size Contribution</th>
<th>% Contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cochlear Filter Units</td>
<td>30</td>
<td>1839</td>
<td>55,170</td>
<td>&lt;0.1%</td>
</tr>
<tr>
<td>Hair Cell Units</td>
<td>30</td>
<td>2789</td>
<td>83,670</td>
<td>0.1%</td>
</tr>
<tr>
<td>Auditory Nerve Fiber Units</td>
<td>30x40x20~24,000</td>
<td>808</td>
<td>19,392,000</td>
<td>32.9%</td>
</tr>
<tr>
<td>Dendrite Filter Units</td>
<td>24,000</td>
<td>634</td>
<td>15,216,000</td>
<td>25.8%</td>
</tr>
<tr>
<td>Stellate Cell Soma Units</td>
<td>24,000</td>
<td>1,708</td>
<td>40,992,000</td>
<td>69.5%</td>
</tr>
<tr>
<td>Coincidence Cell Units</td>
<td>600</td>
<td>111</td>
<td>666,000</td>
<td>1.1%</td>
</tr>
<tr>
<td>Total Gate Size</td>
<td></td>
<td></td>
<td>58,952,040</td>
<td></td>
</tr>
</tbody>
</table>

Table 6-1: Estimated system size based on one to one approach.

Table 6-1 show the estimated system size based on this approach using the implementation size results of the components from chapter 4 and 5. From the total estimated gate size, it is clear that this implementation approach will be difficult to achieve using present day implementation technology and will not be feasible using a small array of FPGAs. From the percentage contribution to the implementation size in Table 6-1, it is clear that the three components that heavily affect the system size are the stellate cell soma units, dendrite filter units and the auditory nerve fibre units. Their contribution to the system size has to be reduced dramatically to make the implementation of the whole system feasible.

6.2.2 Sharing of Noise and Dendrite Units.

As shown in the architecture of the cochlear channel in Figure 6-1(b), each bank of stellate cells requires 40 independent auditory nerve fibre units and 40 dendrite filter units. This is because the noise entering each stellate cell in a bank must be different in order to desynchronise their spiking response. However, given the tree structure and feed-forward nature of the system, it is possible to share these noise sources between the different banks of stellate cells. This is because noise correlation between the different banks of stellate cell soma units will not cause a problem since their outputs will not be synchronised as a result of the different set of parameters used in
each bank of cell somas. With the sharing of the auditory nerve fibre units, the dendrite filter units connected to the each shared unit can also be shared in the same way since they will be producing the same outputs. Figure 6-2 shows the data flow structure of this system base on sharing the auditory nerve fibre units and the dendrite filter units.

![Diagram of cochlear channel](image)

**Figure 6-2: Data flow structure of a cochlear channel where the auditory nerve fibre units and dendrite filter units are shared.**

<table>
<thead>
<tr>
<th>Component</th>
<th>Component Count</th>
<th>Gate Size (Each)</th>
<th>Total Gate Size Contribution</th>
<th>% Contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cochlear Filter Units</td>
<td>30</td>
<td>1839</td>
<td>55,170</td>
<td>0.1</td>
</tr>
<tr>
<td>Hair Cell Units</td>
<td>30</td>
<td>2789</td>
<td>83,670</td>
<td>0.2</td>
</tr>
<tr>
<td>Auditory Nerve Fiber Units</td>
<td>30x40=1,200</td>
<td>808</td>
<td>969,600</td>
<td>2.2</td>
</tr>
<tr>
<td>Dendrite Filter Units</td>
<td>1,200</td>
<td>634</td>
<td>760,800</td>
<td>1.7</td>
</tr>
<tr>
<td>Stellate Cell Soma Units</td>
<td>30x40x20=24,000</td>
<td>1,708</td>
<td>40,992,000</td>
<td>94.2</td>
</tr>
<tr>
<td>Coincidence Cell Units</td>
<td>600</td>
<td>111</td>
<td>666,000</td>
<td>1.5</td>
</tr>
<tr>
<td><strong>Total Gate Size</strong></td>
<td></td>
<td></td>
<td>43,527,240</td>
<td></td>
</tr>
</tbody>
</table>

**Table 6-2: Estimated system size with sharing of the dendrite filter units and the auditory nerve fibre units.**

Table 6-2 shows the estimated implementation size of such a system. The implementation size is clearly reduced by about 26% using this approach. Most of the
Chapter 6

Pitch System Implementation

contribution to the system now is by the stellate cell soma units. However, at this implementation size, it will still be difficult and very expensive to implement the entire system using today’s integrated circuit technology or on a small array of FPGAs.

6.3 Applying Multiplexing

One of the advantage of the digital implementation approach, as mentioned in Chapter 1, was the ease of multiplexing in digital VLSI. Since the implementation results in Chapters 4 and 5 show that most of the components are capable of operating very much faster than real-time, even when implemented using slow FPGAs, it is possible to use multiplexing to reduce dramatically the number of physical components in the system.

6.3.1 One of Each Component Multiplexing Approach

One possible approach of multiplexing is to implement the entire system using one of each type of component to simulate all the units in the system. Figure 6-3 shows the data flow architecture of such a system.

Since each unit now has to simulate all similar units in the system, the operating speed has to be increased dramatically. For example, if a component has to emulate \( N \) units in the system (a multiplexing ratio of \( N:1 \)), the unit has to operate at an epoch rate of \( N \) times its original epoch rate. In Figure 6-3, the resulting epoch rate of each
component is shown. With multiplexing, the number of data paths is also reduced. However, the data rate of each data path is also increased by the same factor. These data rates are also shown in Figure 6-3.

Since for all components, with the exception of the hair cell units, each epoch cycle requires 32 clock cycles, using this multiplexing approach would mean that the maximum clock rate in the system used by the stellate cell soma unit would be 480 MHz epoch rate times 32 which is 15,360 MHz. This clock rate is far beyond the capabilities of present and up-coming implementation technology. Therefore this approach is not feasible.

6.3.2 Multi-channel Approach to the Whole System

To reduce the maximum clock rate in the system to a feasible level, an alternative approach to multiplexing is to use multiple modules in place of those that previously required very high clock speed. This approach includes partitioning the data flow into a few paths (channels) to share the processing load and therefore dividing up the data rate, epoch rate and consequently, the maximum clock rate. In the pitch system, the components that require high speed of operation when being multiplexed are the auditory nerve fibre units, the dendrite filter units and most of all, the stellate cell soma units. Therefore the data paths should be partitioned just before these components. This partitioning approach is shown in Figure 6-4, where the output from the hair cell unit is partitioned into a number of similar channels.

![Figure 6-4: System layout using multi-channel multiplexing scheme.](image)
Chapter 6 Pitch System Implementation

The speed requirement of the system reduces with the number of channels used. However, increasing the number of channels, will also increase the system size. Therefore, a balance has to be struck between the advantage of speed reductions and system size. Table 6-3 below shows an estimation of how the number of channels used will affect the epoch and clock rate of each component. Table 6-4 shows how the number of channels will affect the system size.

<table>
<thead>
<tr>
<th>Number of Channels</th>
<th>Dendrite Filter &amp; Hair Cell Unit</th>
<th>Auditory Nerve Fibre, Dendrite Filter &amp; Coincidence Cell Unit</th>
<th>Stellate Cell Soma Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Epoch Rate (MHz)</td>
<td>Minimum Clock Rate (MHz)</td>
<td>Epoch Rate (MHz)</td>
</tr>
<tr>
<td>2</td>
<td>0.6</td>
<td>19.2</td>
<td>12</td>
</tr>
<tr>
<td>4</td>
<td>0.6</td>
<td>19.2</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>0.6</td>
<td>19.2</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>0.6</td>
<td>19.2</td>
<td>2.4</td>
</tr>
<tr>
<td>20</td>
<td>0.6</td>
<td>19.2</td>
<td>1.2</td>
</tr>
<tr>
<td>40</td>
<td>0.6</td>
<td>19.2</td>
<td>0.6</td>
</tr>
<tr>
<td>80</td>
<td>0.6</td>
<td>19.2</td>
<td>0.3</td>
</tr>
<tr>
<td>160</td>
<td>0.6</td>
<td>19.2</td>
<td>0.15</td>
</tr>
<tr>
<td>320</td>
<td>0.6</td>
<td>19.2</td>
<td>0.07</td>
</tr>
<tr>
<td>600</td>
<td>0.6</td>
<td>19.2</td>
<td>0.04</td>
</tr>
</tbody>
</table>

Table 6-3: Number of channels versus the minimum required epoch and clock rate of the components to achieve real-time operation.

<table>
<thead>
<tr>
<th>Number of Channels</th>
<th>Cochlea Filter Unit</th>
<th>Hair Cell Unit</th>
<th>Auditory Nerve Fibre Unit</th>
<th>Dendrite Filter Unit</th>
<th>Stellate Cell Soma Unit</th>
<th>Coincidence Cell Unit</th>
<th>Total System Gate Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.839</td>
<td>2,789</td>
<td>808</td>
<td>634</td>
<td>1,708</td>
<td>111</td>
<td>7,889</td>
</tr>
<tr>
<td>4</td>
<td>1.839</td>
<td>2,789</td>
<td>8,080</td>
<td>6,340</td>
<td>17,080</td>
<td>444</td>
<td>17,672</td>
</tr>
<tr>
<td>10</td>
<td>1.839</td>
<td>2,789</td>
<td>16,160</td>
<td>12,680</td>
<td>34,160</td>
<td>2,220</td>
<td>69,648</td>
</tr>
<tr>
<td>20</td>
<td>1.839</td>
<td>2,789</td>
<td>32,320</td>
<td>25,360</td>
<td>68,320</td>
<td>4,440</td>
<td>135,068</td>
</tr>
<tr>
<td>40</td>
<td>1.839</td>
<td>2,789</td>
<td>64,640</td>
<td>50,720</td>
<td>136,640</td>
<td>8,880</td>
<td>265,906</td>
</tr>
<tr>
<td>80</td>
<td>1.839</td>
<td>2,789</td>
<td>128,280</td>
<td>101,440</td>
<td>273,280</td>
<td>17,760</td>
<td>528,388</td>
</tr>
<tr>
<td>160</td>
<td>1.839</td>
<td>2,789</td>
<td>256,960</td>
<td>202,880</td>
<td>546,640</td>
<td>35,520</td>
<td>1,048,148</td>
</tr>
<tr>
<td>320</td>
<td>1.839</td>
<td>2,789</td>
<td>484,800</td>
<td>360,400</td>
<td>954,800</td>
<td>68,600</td>
<td>1,981,228</td>
</tr>
</tbody>
</table>

Table 6-4: Number of channels in the system versus the system size.

Based on present ASIC implementation technology, it will be possible to achieve very high operating clock speed using the bit-serial approach shown in Chapter 4 and 5. Speeds of greater than 192 MHz should be feasible. Therefore a system based on no more than 80 or even 40 channels can be achieved. When using 80 channels, the implementation size reaches around 266 K.gates. This implementation size is also well within the capabilities of present ASIC implementation technology.
However, if an FPGA implementation approach is considered instead, a much lower clock speed of typically less than 40 MHz has to be used. This reduction of the speed is due to the much longer delays resulting from the limitations of programmable routing and logic blocks inherent in FPGAs. In the implementation work detailed in Chapters 4 and 5, we were only able to achieve a typical clock speed of around 32 MHz using the Xilinx XC4025PG233-6 FPGA. Therefore, many more channels have to be used. For example, at 600 channels, the clock rate will reduce to 25.6 MHz. However, the system size will increase to over 1.9 million gates. At this size, a very large array of FPGAs will be needed to implement the entire system.

Most of the components in the system use variables that need to be stored. This has so far been ignored in the analysis of system size and operating speed when using multiplexing. Multiplexing is able to reduce the number of physical components needed. However, it will not be possible to reduce the amount of memory needed to store the variables. These have to be stored either in registers, or more efficiently in terms of size using random access memory (RAM). Table 6-5 shows the amount of memory needed per component and for the entire system based on the 16 bit bit-serial approach used in Chapters 5 and 6. Each variable is 16 bits wide except for the spike variables in the stellate cell soma unit which can be represented by one bit each.

<table>
<thead>
<tr>
<th>Component Name</th>
<th>Component Memory Requirement (bits)</th>
<th>No. Of Cells In The System</th>
<th>Total Contribution To System (bits)</th>
<th>% Contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cochlear Bandpass Filter Unit</td>
<td>3x16 = 48</td>
<td>30</td>
<td>1,440</td>
<td>0.16%</td>
</tr>
<tr>
<td>Hair Cell Unit</td>
<td>3x16 = 48</td>
<td>30</td>
<td>1,440</td>
<td>0.16%</td>
</tr>
<tr>
<td>Auditory Nerve Fibers Unit</td>
<td>1x16 = 16</td>
<td>1,200</td>
<td>19,200</td>
<td>2.19%</td>
</tr>
<tr>
<td>Dendrite Filter Unit</td>
<td>2x16 = 32</td>
<td>2,400</td>
<td>38,400</td>
<td>4.38%</td>
</tr>
<tr>
<td>Stellate Cell Soma Unit</td>
<td>2x16+2x1 = 34</td>
<td>24,000</td>
<td>816,000</td>
<td>93.1%</td>
</tr>
<tr>
<td>Coincidence Cell Unit</td>
<td>0</td>
<td>600</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>876,480</td>
<td>100%</td>
</tr>
</tbody>
</table>

Table 6-5: Memory requirements for each type of component and for the entire system.

The ideal approach to implementing the 110 Kbytes of memory would be to integrate it on the same chip as the rest of the system, as registers or RAM. This would allow
the memory to operate at the same speed as the system. However, this approach will also dramatically increase the cost of implementation.

An alternative but potentially less costly approach is to use cheaper, standard external memory devices to implement the memory. For this implementation approach, the issue of the access speed of the external memory used will also affect the choice of the minimum number of channels in the system. For example, if high speed 5 ns access time memories are used, assuming that it takes 32 clock cycles to read and store 16 bit variables serially for each component during each epoch (i.e., the memory operates at the same clock rate of the component), the maximum clock rate will be 200 MHz. This equates to around 80 channels in the system.

Another issue that will determine if the external memory approach will be feasible is the number of data lines needed to the memory. If too many data lines are needed, it may be physically impossible to connect the memory devices to the chip. To calculate the estimated bus width of the memory needed in the previous example, we first calculate the size of data each data line of the memory can handle per sample period. This is equal to 200 MHz clock rate divided by 20 kHz sample rate which equals 10,000 bits. Since each bit has to be read and updated per epoch, this reduces to 5000 bits per sample period. In total, 876,480 bits need to be stored per sample period, the estimated minimum number of data lines or pins needed is found by dividing 876,480 bits by 5000 bits, which results in 176 pins. These data pins along with the 13 address lines needed for addressing the 5000 word locations would add up to 189 lines. This number of signal lines or package pins is well within the limits of present packaging technology.

After evaluating the multi-channel approach to implementation, it is clear that the size and speed of the system allows a single digital VLSI integrated circuit with external memory devices to be used. However at the rate of advance in implementation technology, providing higher densities and speed, it will be possible in the very near future also to implement the required memory on the same chip cost-effectively, resulting in a more efficient design. In the thesis, none of the approaches just discussed was taken because of funding limitations.
6.3.3 Single Cochlear Channel Approach

Because of the limits on funding in this work, we were forced to look for an alternative to allow us to experiment on a representative part of the system with the possibility of implementing onto an FPGA and scaling it up to the complete system if needed in the future. One approach is to implement a single cochlear channel instead of the full system which will reduce the implementation size to a thirtieth of the full system. This single cochlear channel will be able to provide a representative section of the system for test because all cochlear channels are the same except for the pass-band frequencies of the cochlear filters. The frequencies can be changed to correspond to any channel by varying the parameters of the system. Figure 6-5 shows the data flow architecture of such an implementation incorporating multiplexing.

![Data flow architecture of a single cochlea channel of the Four Stage Pitch System with multiplexing.](image)

A multiplexing ratio of 1 to 40 is chosen based on the implementation results detailed in Chapter 4. This multiplexing ratio also allow us to multiplex 40 stellate cell somas in a bank using a single physical unit, allowing each unit to be programmed only with one parameter throughout its operation. The coincidence cell on the other hand is not multiplexed. This is because multiplexing it will not make substantial savings in gate size since there are only 20 units, each at only 111 gates, making a total of 2220 gates.
The expected gate size of the implementation is around $43000$ gates which will not pose a problem for modern FPGA technology. The total number of data lines needed for the memory will also not pose a problem for present integrated circuit (IC) packages since the memory requirement is now only $1/30$ of the full system. Even in the worst case, where each component has its own independent data path to the memory device, it will only need around $69$ data lines when the memory is operated at the clock rate of $25.6$ MHz.

Figure 6-6 shows the expected hardware structure of the system. In the diagram, the input sound stimulus is converted into digital sample data by an analogue to digital converter (ADC). The memory banks contains the external memory devices needed for the storage of the variables. The programming, timing and control block contains necessary logic for the programming of the FPGA, loading of parameters and providing clocking and other control signals. Finally, the FPGA contains the entire cochlear channel implementation along with memory control circuitry. This unit will produce an array of $20$ output spiking signals, each corresponding to a pitch that the system needs to detect.

To scale this up to the complete system, all that is needed will be to build more identical cochlear channel units but program these with different parameters. If
implementation technology allows, it is also possible to implement a number of these cochlear channel units onto a single chip, reducing the logic circuit overheads needed for driving the memory.

### 6.4 Implementation

To test if the system architecture will work as expected, the system has to be modelled, implemented and tested. Two approaches were used. The first involves implementing the system in VHDL and then into an FPGA. The second approach involves simulating the system using a LabView based model of the digital hardware.

#### 6.4.1 VHDL and FPGA Implementations

All components were first implemented using VHDL. Their behaviour was subsequently simulated and tested to check that they performed as expected. The VHDL code was then synthesised into a circuit netlist. This is again simulated to determine if it matches the VHDL behavioural simulation. Finally the netlist of each component is placed onto a FPGA and tested to verify that the circuit will work. The results of this implementation work were discussed previously in Chapter 4.

In building the single cochlear channel system, we start off with the tested VHDL description of the components. These are slightly modified to handle multiplexing and then put together to form the system. The system is then simulated to determine if it is performing correctly. Because of limited availability of software design tools and funding, we were only able to implement up to two components onto a single Xilinx 4013PG-233FPGA at a time. Therefore, the system is verified in hardware by testing each component in the system individually, but making sure that the data stream from each stage is correct and able to feed to the next stage. This method will allow us also to test the speed of the system, to prove that it will indeed be possible to provide real-time performance.
6.4.2 LabView Based Implementation

The tools and funding limitations do not allow us to implement the single cochlear channel in actual hardware. Therefore, to test the behaviour of the system, we were forced to simulate it. However, VHDL simulation was found to be very slow. Typically, simulation of a single cochlear channel for a real-time period of half a second (10,000 input samples) will take an estimated time of around 2 months on a Sun Sparcstation 10 workstation. Therefore an alternative is to construct a more efficient software simulation of the system. The software model has to mimic the arithmetic operation of each component accurately so that behaviour will match the proposed FPGA implementation of a single cochlea channel.

We choose to implement the software version of the system using Labview which is an object-oriented graphical programming language that provides a fast route to program design. Charting and graphical modules built into the language make it possible to visualise the results quickly. Using a Labview simulation of the system, a Intel based PC with a 486DX2-66 processor will take around two days to complete the same simulation mentioned above. This will therefore be used instead in the experiments detailed in the next chapter to test the behaviour of the single cochlear channel of the FSPS.

6.5 Summary

This chapter has shown how the entire system can be structured in hardware based on the components mentioned in Chapters 4 and 5. It has also shown how the auditory nerve fiber units and dendrite filter units can be shared to reduce the number of components in the system. Multiplexing and the use of external memory modules was then proposed as the only way to reduce the gate size of the system to make it feasible for the complete system to be implemented using present day semiconductor technology.
This chapter has also proposed an alternative approach where a representative part of the system, a single cochlea channel, is implemented. The size of this system will allow it to be implemented onto a single FPGA with external memory modules. The software simulation of such a representative system will also be sufficiently fast for experimentation.

In conclusion, the chapter has shown that it is possible to implement a representative part of the complete FSPS in hardware and to operate it in real-time.
Chapter Seven
Pitch System Test and Analysis

7.1 Objectives of Chapter

In this chapter, we look into the Labview based experimentation on the sub-system of the FSPS, the cochlear channel. The objectives of this chapter are:

- To present the simulation set-up and the results of the experiments.
- To analyse the results to determine if they meet the expected behaviour of the system.
- To discuss the possible reasons for areas where the system fails to meet expectations.

7.2 Test System Set-up

As mentioned in the previous chapter, a single cochlear channel of the FSPS was implemented in software using the Labview graphical programming language. The simulation used bit-limited arithmetic and simulated the arithmetic process and the data flow structure of the system. Only one representative cochlear channel was implemented in software because of the limit on computing power available to us.

In the following section presenting the individual experiments, the stimulus files needed for each simulation were pre-generated and supplied to the software model as a text file. The simulation then placed the resulting output of the system into an output file. For each simulation, the different bandpass frequencies of the cochlear filter along with the gain values were set by varying their parameters. The 0 dB input level
for the system was determined by adjusting the system gain in such a way that the system just starts to produce a response to the input stimulus. Higher input stimulus intensity was provided by increasing the input gain at the input from the reference 0 dB level.

For each simulation, the resulting data were analysed by plotting the output spike count of each coincidence cell in the system as shown in Figure 7-1. Since each coincidence cell connects to a bank of stellate cell of a specific best modulation frequency, if a coincidence cell output has a relatively high spiking rate as compared to the rest, it shows that it is phase locking to the input stimulus modulation. The most active spiking coincidence cell then denotes the pitch of the input stimulus. For simplicity, we identify the coincidence cell which is connected to a bank of stellate cells of a specific best modulation frequency by labelling the coincidence cell with the same best modulation frequency. In the system, the best modulation frequencies of the stellate cells are set at an input intensity of 30 dB by varying their parameters.

![Detected Pitch Frequency](image)

**Figure 7-1: Typical plot of the simulation outputs.**

All simulations in this chapter use stimulus files with 10000 samples, an equivalent of half a second duration in real-time. As a result, the output spike count of each coincidence cell can be considered simply as the half the average spike output rate recorded. Usually, when a coincidence cell has maximum phase locking with the modulated stimulus, the average spike rate will be close to the modulation frequency, depending on the coincidence cell threshold level. However, it is not the absolute spike rate that is important, but rather the ability to distinguish pitch by showing the highest relative spike rate. The coincidence cells’ threshold value in the system was
therefore set to optimise the results to give clear peaks. This threshold value, once chosen, was used as standard on all coincidence cells throughout each experiment.

7.3 Simulation Results and Analysis

In the following sub-sections, the results of a number of tests will be discussed and analysed to determine the behaviour of the FSPS implementation and its correlation with Hewitt and Meddis's software simulation results [Hewitt94].

7.3.1 Pure Tone Test

In the first experiment, pure tones with frequencies higher than 1 kHz were used as the input to the system with various input intensities. For each pure tone, the cochlear bandpass filter unit was set to the same centre frequency as the input tone to minimise any attenuation. This test was performed to test the response of the system to tones which are much higher than the pitch frequencies that it is capable of detecting and which contain no modulation information. Since there is no modulation in the stimulus, the stellate cells are not expected to phase lock to the input stimulus. Therefore, no coincidence cells in the experiment is expected to show a peak relative to the rest of the coincidence cells in the cochlear channel especially at high input tone frequencies. Figure 7-2 shows the results of the experiments at various frequencies and intensities with the coincidence threshold level set to 8.
Figure 7-2: Pure tone input test at different frequencies and intensity.
From the results, it is clear that at very high input frequencies of 5 kHz, none of the coincidence cell units shows much higher spiking rate compared to neighbouring cell units at any intensity as expected. However, at lower input frequencies of around 3 kHz and 1.8 kHz, some coincidence cells start to show some signs of phase locking especially at higher input intensities. Fortunately, the average spike rates of these coincidence cells at 30 dB input intensity are small compared to their best modulation frequencies. At just over 1 kHz, the output shows phasing locking at a number of coincidence cells. The most likely reason for these peaks is that the dendrite filter units in the system are only designed as first order lowpass filters, which do not have sufficient attenuation for frequency components near to the cut-off frequency. These unwanted frequency components are therefore strong enough to trigger phase locking characteristics at the stellate cell soma units. This is supported by the fact that at 30 dB intensity, the spike rate peaks occurs at cells that have best modulation frequency close to a integer fraction of the tone frequency. For example in Figure 7-2c, the 30 dB spike count peak occurs at around a tenth of the tone frequency while in Figure 7-2d, the 30 dB spike peaks occurs at around 1/4, 1/5, 1/7 and 1/8 of the tone frequency. At higher input intensities, the best modulation of the stellate cells varies slightly resulting in the small shift of the spike count peaks.

Since the Hewitt and Meddis FSPS model was based on physiological data from small animals where their hearing is limited to a lowest frequency of greater than 1 kHz, it is therefore reasonable to disregard the results of pure tone signals near 1 kHz. As a result, almost all experiments in their papers used 5 kHz as the carrier tone frequency. Therefore, we can conclude that in this experiment, the system was operating correctly. However, it is possible to improve these by implementing the dendrite filter units using higher order low pass filters. This will reduce the intensity of high frequency components at the stellate cell soma units, reducing the unwanted phase locking characteristics.

7.3.2 Random Noise Test

The next experiment involves using random noise as the input stimulus. These were performed to test the response of the system to a source of input which can be
considered as a collection of randomly modulated tones. When the random noise is passed through the cochlear filter, it reduces to a single randomly modulated pure tone with the pure tone frequency corresponding to the cochlear filter's centre frequency. Since the modulation of the pure tone is random, none of the coincidence cells is expected to show a significantly higher spike count than others. However, it is possible that each of stellate cell soma banks may phase lock for short periods with the modulation rate when it is momentarily the same as the BMF. This should occur with all stellate cell soma banks, resulting in an overall increase in the spike count. Figure 7-3 shows the results of the experiments where noise of three different intensities was used with the cochlear bandpass frequency set to a number of representative values.

![Diagram](image-url)
From the results in Figure 7-3, it is clear that the overall spike count has increased as compared to the pure tone test results. The spike count values are relative to the input intensity and also to the best modulation frequency of the coincidence cells. The reason why the spike count rate is proportional to best modulation frequency is that the output spike rate of the coincidence cells with higher frequency will produce more spikes in the same periods as compared to cells with lower frequencies. At 5 dB and 30 dB input intensity, there are no large peaks, indicating that the system is not detecting any particular pitch. However, at the higher input intensity of 60 dB, when higher cochlear bandpass filter centre frequencies are used, some spike count peaks can be seen. This can be attributed to the phase locking characteristics of the system to the carrier frequency of the filtered noise, similar to the effects found when using pure tone signals with frequencies near to the cut-off frequency.

From the results of this set of experiments, it is clear that the system largely performed according to expectation. At high input intensities and high cochlear filter centre frequencies, the system did not performed as well because of the phase locking of the stellate cell soma banks to the carrier frequencies. However, this did not adversely affect the expected performance of the system in this test.
7.3.3 Modulated Tone Test

This experiment involves testing the system with a fixed intensity pure tone signals that are modulated at various frequencies. This experiment was performed to test the ability of the system to detect the modulation (pitch) of the input stimulus. The success of this experiment is critical in verifying that the system is able to perform the task of pitch detection that it is designed for.

In these experiments, a number of representative carrier signals was used. Each carrier signal was modulated at 25% at a number of representative frequencies covering the pitch range that the system is designed for. For each experiment, the cochlear bandpass filter was set to the same centre frequency as the carrier tone so as to reduce any attenuation. All coincidence cell thresholds were set at a level of 8. Since the system is not designed to detect modulation with carrier tone frequencies less than 1kHz, the representative carrier signal frequencies used will only start at 1kHz. Figure 7-4 shows a representative selection of the result of this test with the input intensity set at 30dB and with a number of modulation frequencies.
b) Tone Frequency at 1838Hz

- 106Hz
- 140Hz
- 184Hz
- 212Hz

Coincidence Channels

Spike Count

C) Tone Frequency at 3350Hz

- 106Hz
- 140Hz
- 184Hz
- 212Hz

Coincidence Channels

Spike Count
Chapter 7

Pitch System Test and Analysis

The results in Figure 7-4 clearly indicated that at high tone frequencies, for example at 5kHz, the system was able to detect the modulation frequency or pitch of the input signal by showing spike count peaks at the relevant coincidence cells. For example in Figure 7-4d the tone modulated at 184 Hz caused the output to show the highest spike count peak at the 184 Hz coincidence cell. However, as the tone frequency decreases, the performance of the system degrades.

As clearly seen in Figure 7-4a and 7-4b, the results show that at these lower tone frequencies, the effect of phase locking to the higher order frequencies of the carrier itself is disrupting the detection of the modulation. Again, this can be attributed to high frequency components that remain after filtering by the dendrite filter units. Another interesting effect that can be noticed in the results is that with low tone frequencies and low modulation frequencies, coincidence cells with best modulation rate at multiples of the tone’s modulation are also showing high spike counts. This is most likely the result of phase locking by the stellate soma banks that have best modulation frequency which are integer multiples of the stimulus modulation rate. However we do not yet understand why this is reduced at higher tone frequencies.

Figure 7-4: Response to 30dB modulated pure tones.
From these results, we can see that the system is able to detect modulation information clearly, but only when the tone frequencies are much higher than 1 kHz. The 5 kHz carrier tone results show that the behaviour matches Hewitt and Meddis results even though the amount of modulation used was smaller. As mentioned before, some of the interference problems caused by phase locking to high frequency components can be again reduced by improving the dendrite filter unit implementation using higher order filters.

### 7.3.4 Modulated Tone Intensity Test

In this test, modulated tones at different input intensities were used as the input into the system. This was performed to test how the change in intensity will affect the system's ability to detect modulation. The best modulation frequencies of the stellate cells were set at 30 dB input stimulus intensity. At this input intensity, the system should perform as expected, detecting the modulation or pitch of the input stimulus. But at high input intensity, the modulation of the input signal to the system is reduced because of the saturation function in the hair cell implementation. Therefore, this is expected to reduce the capability of the stellate cells to phase lock to the modulation tone. Since the modulation rate we have chosen is very low at 25%, at high input intensities, the stellate cells might not be able to detect the modulation at all. At very low intensities, the stellate cells are expected to have very low spike rates and as a result, should not show much phase locking to the stimulus. At low intensities, the best modulation frequency of the stellate cells will also decrease, such that if there is any phase locking, it will only occur at cells originally set to detect higher modulation frequencies.

For this test, a 25% modulated tone at 3350 Hz was used as the input stimulus. Three different intensities of 5 dB, 30 dB and 60 dB were used, along with a number of representative modulation or pitch frequencies. The cochlear filter's centre frequency was set to 3350 Hz to reduce attenuation. Figure 7-5 shows some representative results of this test.
Figure 7-5: Representative results of amplitude modulation intensity test.
From the results in Figure 7-5, we can clearly see that at 30 dB the system performed largely as expected, detecting the correct modulation or pitch frequency of the input stimulus. When the system was stimulated by high intensity signals, it is clear from the results that the system loses its ability to detect clearly the modulation information as expected. This is consistent with the results found by Hewitt and Meddis. At low input intensity, we can see from the results that the modulation information is detected by coincidence cells with higher best modulation frequency cells, and there is not a very sharp peak as seen in Figure 7-5 (a) and (b). This shows that the stellate cells at very low intensity inputs will have phase locking characteristics to all modulation frequencies below a point which is lower than the best modulation frequency of the stellate cells at 30 dB. Again, this is consistent with the results found by Hewitt and Meddis, who stated that the best modulation frequency of the stellate cells are intensity dependent and, at very low input intensities, the stellate cells shows a low pass type modulation transfer function. Therefore at high intensities and very low intensities, the capability of the system to detect modulation information is severely affected. Since the system performed as expected, we can conclude that the system’s behaviour in this test is correct.

7.4 Further Analysis of Results

From the experimental results in this chapter, we can see that the system performed well at the task of pitch detection at 30 dB input intensity and when high carrier tone frequencies are used. Without modulation, tones of high frequencies are ignored by the system. This matches the behaviour of the Hewitt and Meddis FSPS simulations.

Hewitt and Meddis, however, did not perform any experimentation using lower frequency carrier tones. Consequently, it is not possible to compare these results which shows that at lower frequency pure tones and carrier tones, the stellate cells phase locks to either the input stimulus carrier frequencies or to the higher order harmonics of the carrier frequencies. This effect increases as the tone frequency
Chapter 7 Pitch System Test and Analysis

decreases and can cause the system to fail to detect the modulation information as shown in Figure 7-4a. The result may be that the system will perform poorly with human speech because much of the speech bandwidth is below 3 kHz. This problem, we believe, is not a result of an implementation fault but rather a limitation of the original dendrite model, where the first order lowpass filter was used.

A way of improving this system in the future would be to implement the dendrite filters using higher order filters instead of the first order filters used by Hewitt and Meddis which we have chosen to follow. This should reduce the amount of higher frequency components reaching the stellate cell soma, delaying the start of this unwanted effect to a lower frequency. This can be interpreted biologically as the dendrite fibres of the stellate cell being longer physically.

Another effect that we have noticed that did not match Hewitt and Meddis' results well is the slight shift of the best modulation frequency of the coincidence cells when the input stimulus is increased from 30 dB to 60 dB intensity, clearly seen in Figure 7-2c. This is the result of a slight decrease of the output from the hair cell after adaptation despite the increase of input stimulus. We have no clear indication why this is happening, but suspect that either the bit-width resolution for the hair cell is still not sufficient and/or the parameters of the hair cell unit needed some adjustments. This problem, however, did not cause the system to fail to meet the important system behaviour required.

7.5 Conclusion

In this chapter, we have presented the Labview based setup used to test the system. The results show that, in many ways, the system behaviour matches the results in Hewitt and Meddis' paper which the pitch system implementation was based on. This exercise shows that the arithmetic and communication structure of the system is capable of reproducing the important functionality needed. This proves that it is
feasible to implement the system in digital VLSI hardware to reproduce the required system behaviour.

Through these experiments, we have however discovered that the system fails to work properly when lower frequency carrier tones are used. This we identified is caused by use of first order lowpass filters as the dendrite filter which we believe is a limitation inherent in the original model itself. The slight shift of best modulation frequency from 30 dB to 60 dB of the coincidence cell also does not match Hewitt and Meddis results. We believe this could be caused by insufficient bit-width resolution of the hair cells or the need for better hair cell parameters.
Chapter Eight
Conclusion

8.1 Objectives of Chapter

This chapter concludes this thesis and has the following objectives:

• Review the objectives set out in the beginning of the thesis and in Chapter 3
• Summarise the conclusions of the investigations detailed in the previous chapters.
• Determine how successful is the work presented in this thesis.
• Suggest future investigations to extend this work.

8.2 Review of Objectives

To tackle the lack of neuromorphic digital implementations of the brainstem structure for sensory perception, the following aims were set for this research:

• To understand how much of the biological system structure, complexity and components are needed to capture effectively the capabilities of biological systems.
• To know how neuromorphic components and systems can be implemented efficiently in digital hardware.

Since the initial issue that has to be tackled to work towards these aims is feasibility, we therefore chose to implement a representative system based on the FSPS to determine this. As a result, the following objectives are set for the work in this thesis:
• For the system to be feasible, models have to be found for each component of the system that can reproduce the necessary behaviour while keeping the implementation size down.

• In digital systems, discretisation is unavoidable. Therefore, the resolution and accuracy requirements of the model have to be investigated. A resolution must be chosen for the system that balance the implementation size and complexity with the functional requirements.

• The FSPS is made up of a large number of components. Therefore the architecture of the system has to be investigated with the objective of reducing the number components and communication lines used.

• The size of the implementation has to be investigated since it will determine how realistic such an implementation will be. The size of the implementation has to be within the limits of present and up-coming technology.

• The complete system has to be able to reproduce the required behaviour set out in the original model. Therefore, the behaviour of the implemented system has to be determined with the objective of assessing its suitability of fulfilling that role.

Since this implementation work is pioneering, it is essential that experience is gained that can be applied to other similar systems. Therefore, the last two additional objectives are:

• Implementation issues that are critical to this type of system have to be identified.

• A way of dealing with the issues identified has to be proposed.

To achieve these objectives in the implementation of the FSPS, we start our investigations by implementing the individual components in the system. This was followed by investigations into the structure of the entire system to determine if it will be possible to implement this system using present day implementation technologies. Then a software simulation was performed based on the structure and the bit-limited arithmetic expected in the system. This is to determine if the system will be able to reproduce the expected behaviour.
8.3 Conclusions

In Chapter 1 and 2, we have discussed both the lack of neuromorphic digital implementation on the sub-system of the sensory brainstem and also the potential benefits of the digital approach. This resulted in this work of implementing the FSPS in order to test its feasibility.

In Chapter 4, we first looked into the implementation of the components in the FSPS excluding the stellate cell. We were able to show how these components can be implemented so that they will reproduce the important behavioural characteristics required. The implementation results shows that the operating speed is fast enough to achieve real-time operation with the possibility of multiplexing to reduce the component count in the system. These components are also small enough to allow large systems to be built. This chapter meets in part the first objective of this thesis.

Chapter 5 then looks into the implementation of the stellate cell. This chapter shows how this most numerous component in the system can be implemented in two parts, the dendrite filter unit and the stellate cell soma unit. These components are able to reproduce the required functionality and are not only efficient in terms of gate size, but are also able to operate many times the required real-time speed, paving the way for large arrays of these cells to be built efficiently. The resolution and accuracy issue was also investigated. From the results, we choose a bit-width of 16 bits which not only will provide more than sufficient accuracy for the stellate cell soma, but also allows it to be implemented more easily. This chapter meets the first and second objectives of this thesis.

Chapter 6 then looks into the issues relating to the implementation of the complete system. In it, we showed how the entire system is structured in hardware based on the components in the previous chapters. We have also shown how the auditory nerve fiber units and dendrite filter units can be shared to reduce the number of components in the system. Multiplexing and the use of external memory modules was then proposed. A few possible architectural approaches were detailed which include a sub-
system approach. These approaches provide the only way to reduce the implementation size of the system to make it feasible for the complete system to be implemented using present day implementation technology. This Chapter meets the third, fourth and the last two objectives of the thesis.

Finally in Chapter 7, we present the set-up and results of the software simulations performed to test the behaviour of the expected system. This shows that in many ways the system behaviour matches the results in Hewitt and Meddis paper which the pitch system implementation was based on. This proves that the system is capable of reproducing the important functionality needed. This fulfils the fifth objective of the thesis. This along with the previous chapter also proves the feasibility of implementing such a system in hardware.

8.4 Measure of Success

After reviewing the conclusions of the each chapter, it is clear that all the objectives set out in the beginning of the thesis and in Chapter 3 have been met. Specifically, the work in this thesis has resulted in the following:

- In Chapter 2, we state that digital implementations of the auditory sub-systems are few. Therefore, in this work, we were the first to implement in digital hardware a wide range of components that are essential in any digital hardware modelling of the auditory system. These implementations are not only able to reproduce the necessary behaviour, but are efficient in gate size and capable of operating much faster than real-time. This makes it possible for large systems to be built.

- We were also able to show analytically how the system architecture influences the system size. Through this, it was shown that the complete system size can be within the limits of present day integrated circuit technology.

- Through experimentation, we were able to also show that the FSPS implementation was capable of reproducing the required behaviour. This along with the previous
point prove that it is feasible to implement the Four Stage Pitch System in digital hardware. Consequently, the results also give clear indications of the possibility of implementing large neuromorphic signal processing systems in hardware.

- This work has also identified the principles essential in reducing system size. Included in these principles are the use of simple or simplified models, bit-serial approach to reduce communication lines, sharing of modules and multiplexing. These principles can generally be applied to all digital neuromorphic signal processing implementations to reduce implementation size.

The digital approach to neuromorphic implementation has always been viewed by many as less viable compared to the analogue approach [Schiak96b, Smith98]. This work has shown that this is not true and that the digital approach can provide a very effective and fast route to realising systems in hardware. Clearly, as implementation technology improves, the limits set by implementation size will reduce, making it far easier to implement large neuromorphic systems using the digital approach.

8.5 Limitations of Work

This research is limited by the following:

- No attempts were made to implement large sub-systems or the complete system of the FSPS into hardware. This limitations exist as the result of the funding constraint placed on this work thus far. We were limited throughout this work to implementations that will fit into a single small FPGA of 13 Kgates or 25 Kgates capacity.

- The FSPS, chosen as the basis of the implementation work is essentially a feed-forward system which eases the design of the implemented system. This affects the degree to which the implementation can be considered representative of all sensory neuromorphic systems.
The implementation sizes of the components are based on statistics gleaned from the synthesised netlist of their respective VHDL description. Since the synthesis and optimisation processes are not fully under our control, the implementation size statistics will vary when using a different design software tool and when the target technology is different.

The implementation sizes do not take into account the additional routing and logic requirement needed when using FPGAs, which include hardware needed for programming the target device. This will affect the accuracy of the gate size predictions.

All performance characteristics were obtained by simulation. This limitation results from our inability to test hardware at high speed. Furthermore, since the synthesis netlist was not optimised for FPGA, the final speed will usually be faster than that quoted in this thesis. Therefore the performance characteristics are quoted more to give an indication of speed rather than to give a definitive speed measurement.

We decided to use bit-serial techniques in our implementations without first performing an in-depth study of all possible arithmetic approaches. Even though we believe this approach will result in compact implementations while minimising communication lines used, it is not the definitive solution for reducing implementation resources.

8.6 Suggestions for Future Investigations

In this section, we present areas that has been identified for future investigations that can potentially extent the current work. Areas which are strongly related, but are not part of this work, will also be presented for future investigations.
8.6.1 Extensions of Current Investigations

In the work presented in this thesis, a sub-system of the Four Stage Pitch System was not implemented in hardware because of funding limitations. Therefore, when resources are available, the natural progression of this work would be to implement such a system so that tests can be performed to ascertain further the behaviour of system, especially when using real stimuli.

All stellate cells in the simulation start at the same state at the beginning of each run. This can be considered as the worst case scenario. A more suitable start state would be to assign initial randomised membrane and potassium conductance values to each stellate cell. This will help the system to provide a clearer indication of phase locking using shorter stimulus time.

The auditory nerve fibre units used in the system are a simplified representation of what happens in biology. Even though the system performed as expected, the actual impact of using this approach was not analysed. Therefore, the effect of noise, including the type and level can be investigated to determine its impact on the system operation.

The coincidence cell unit implementation is based on a simple sum and threshold system. However, in Hewitt and Meddis software simulations, a more accurate software implementation based on the MacGregor cell model was used. Therefore, this approach can be investigated to determine if the system size increase can be justified by any potential performance gain.

The hair cell's saturation function, which includes a division function, requires many more clock cycles for each division operation as compared to the rest of the system. Future work could look into alternative ways of implementing this function to increase speed and possibly reduce gate size.
In Chapter 7, we mentioned that phasing locking to low pitch stimulus was occurring due to the nature of the dendrite filter which is based on a first order filter. Future implementation could investigate using higher order filters to improve the filtering characteristics of this unit.

Since the stimuli reaching the stellate cell somas are low pass filtered with a cut-off frequency much lower than the sample rate, a possible way of reducing component count exists through reducing the sample rate of the system after dendrite filtering. Future investigations can look into this option to determine how this will affect the behaviour of the system.

8.6.2 Related Investigations

Only the components required in the FSPS are implemented in this work. However, there are some other common components found in the brainstem which are not covered in this study. These include the onset bushy cells, and Type I cells. The existence of these components may help in other system implementations where they are needed. Future investigations can look into these missing components.

The FSPS was chosen as the basis of this implementation exercise. However, there are other systems in the auditory brainstem that can be implemented based on similar principles and can potentially be beneficial in terms creating a more complete brainstem system in digital hardware. An example is the localisation neural circuits described in Chapter 2.

The principles introduced in this chapter for minimising implementation size can also benefit other sensory system implementations. Future investigations could look to other more complex systems like vision and olfaction, where similar principles and even components can be applied.
References


<table>
<thead>
<tr>
<th>Reference</th>
<th>Author(s)</th>
<th>Title and Details</th>
</tr>
</thead>
</table>
References


References


References


Appendix:

Publications


[1] Digital Realisation of the Mammalian Hair Cell

Digital Realisation Of The Mammalian Inner Hair Cell


Department of Electronic and Electrical Engineering
Loughborough University
Loughborough Leics, LE11 3TU

1Department of Psychology
Essex University Colchester Essex, CO4 3SQ

Abstract

We present the first digital electronic implementation of a key component in the mammalian auditory brainstem - the hair cell. It is implemented in FPGA technology and operates in real-time. Details of cell structure, complexity, speed and performance are given.

1 Introduction

1.1 Motivation

There is substantial interest in developing electronic systems which mimic the smart processing found in living creatures. Implementing the signal processing found in the auditory brainstem has immediate applications in emulation units for physiologists and auditory modellers. There are longer term applications in advanced hearing aids and speech recognition systems. To date most of the work in this area is in analogue implementations. However, advances in microelectronic technologies, allied with ease of design, make digital implementations an attractive approach.

There are 2 immediate possibilities that come to mind: visual and auditory systems. Of these two auditory processing seems the more tractable given current technology and it is on this that our research efforts focus. Specifically, we look at a known phenomenon of the early stages of the auditory brainstem - the capacity to detect the fundamental pitch of a signal. Work on this topic is well underway at Loughborough and Essex and this paper reports on 1 key component of this system - the inner hair cell which transduces audio signals into electrical pulses and also acts as a half-rectifier with automatic gain control properties. Furthermore, as the transducer of the system, it is likely to be needed in all implementations of the auditory brainstem.

Prior to our work, the most advanced hardware implementation of an inner hair cell model was that of Van Schaik [3]. As well as being an analogue implementation, his system model implements only highly-simplified characteristics of the most generally accepted model of the hair cell - the Meddis cell model [1, 2]. In particular, it omits the internal chemical (transmitter) flows within the cell. Both a more sophisticated saturation/adaption mechanism as well as, transmitter flows are necessary for realistic modelling of the Meddis cell model. Portions of the auditory system have also been implemented in analogue hardware by Lyon [4] and more recently by Kumar et al [5]. The design to be described faithfully implements these characteristics.
1.2 Goal of work

The long-term goal of the work is to understand how to design efficient and effective digital implementations of the signal processing within the auditory brainstem. We can also use these models to understand and explore the actual operation of the cells and collections of different cells in real time.

Our first design project aims to produce a pitch extraction system based on neuromorphic principles. This system can take as input a sound signal and resolve the pitch of the signal with an accuracy of 10Hz over a 80Hz-300Hz range (this corresponds to the pitch of the human voice). Pitch is perceived rather than measured and can be related to the fundamental harmonic or amplitude modulation frequency of a sound signal. We are using multiple FPGAs as our implementation technology and VHDL and logic synthesis as the design tools.

The first stage of this work is to construct the component cells found in the auditory brainstem. In this paper we report on the design of one of the cells the 'Inner Hair Cell'. An array of inner hair cells is one of the major components in the auditory periphery. The purpose of the inner hair cell is described later on in this paper.

1.3 Structure of paper

In Section 2 we give an overview of the signal processing that occurs within the auditory brainstem. This starts from the external ear and goes through to the neural nuclei within the auditory brainstem. The Meddis inner hair cell model is detailed in Section 3. We describe the model and our motivation for using it. Sections 4 discusses implementation issues, including arithmetic and data representation approaches together with a consideration of architectural issues. Section 5 reports on experimental results where we measure the response of the hair cell model when we apply a tone of varying intensity to it. We will go on to compare the hardware design performance with biological data. These are results from the first digital implementation of the inner hair cell.

2 Auditory signal processing

The auditory system is split into the auditory periphery and the auditory brainstem. The auditory periphery is the mechanical 'front end' to the auditory brainstem. The auditory brainstem is the neural section.

2.1 Auditory periphery

The auditory periphery comprises the outer, middle, and inner ear. The outer ear comprises the pinna, concha, and ear canal. These features act as filters to the acoustic signals entering the ear.

![Auditory periphery](image)

Figure 1: Auditory periphery.

The middle ear is made up of the tympanic membrane and three ossicles. These ossicles are attached to the cochlea via the oval window. The cochlea contains the inner hair cells which are connected to the auditory nerve (AN) fibres. These inner hair cells are fundamental to the electronic implementation of the auditory pathway as these cells are responsible for converting mechanical vibrations into electrical impulses.

2.2 Auditory brainstem

The auditory brainstem receives its input from the auditory nerve. As mentioned in the previous section, the AN is stimulated by inner hair cells which reside in the cochlea. Each AN fibre is finely tuned to a certain frequency band. The auditory brainstem comprises a number of nuclei,
each of which contain different types of cells and perform different functions.

These nuclei perform functions such as sound source localisation and auditory streaming. We are aware of another system within the brainstem which is the pitch extraction system. This system extracts the fundamental frequency from speech which is made up from a large range of frequency components. The pitch frequency can either the fundamental harmonic or the modulation frequency of a speech signal.

The most complex cell in this system is the hair cell within the cochlear. The job of this cell is to convert the mechanical vibrations of sound into an electrical signal. This electrical signal is transmitted down the AN to the neural centres.

3 The Meddis hair cell model

The inner hair cell in the mammalian cochlea is a key component in the process of hearing. It converts acoustic vibrations into an electrical signal which is used to drive action potentials (spikes) via the auditory nerve to the brain. The system also processes the signal so that the electrical output is quite unlike the mechanical input.

\[
\begin{align*}
(1) \quad k(t) &= \begin{cases} 
\frac{g \cdot dt[s(t) + A]}{s(t) + A + B} & \text{for } [s(t) + A] > 0 \\
0 & \text{for } [s(t) + A] \leq 0
\end{cases} \\
(2) \quad \frac{dw}{dt} &= r c(t) - x w(t) \\
(3) \quad \frac{dq}{dt} &= x(1 - q(t)) + x w(t) - k(t) q(t) \\
(4) \quad \frac{dc}{dt} &= k(t) q(t) - l c(t) - r c(t)
\end{align*}
\]

Figure 2: Meddis inner hair cell model equations.

The signal is half-wave rectified and subject to adaptation (like an automatic gain-control with two parallel time constants). The output saturates early so that the dynamic range of the unit is severely reduced in most cases.

Although the input/output function is highly nonlinear, the system behaviour can be modelled to a first approximation by a small set of differential equations (see Figure 2) [1, 2].

The hair cell releases a transmitter substance when stimulated by sound. This transmitter is released into a synaptic cleft and stimulates auditory nerve fibres. These then respond by transmitting voltage spikes to the auditory brainstem. A spike is a brief and sudden increase in voltage potential. The spike rate is a function of the amount of transmitter in the synaptic cleft which in turn reflects (nonlinearly) the level of sound intensity.

The behaviour of the cell can be modelled in terms of the flow of transmitter substance between three putative reservoirs. One reservoir, the free transmitter pool, holds the transmitter inside the cell ready to be released into the cleft in response to acoustic stimulation. The transmitter in the cleft is represented by the second reservoir and it is this amount that determines the rate of spike activity in the auditory nerve fibre. The amount of transmitter in the cleft can be considered as the probability of the cell transmitting a spike at any one time. Transmitter in the cleft is taken back into the cell and reprocessed in the third reprocessing reservoir. The system is represented diagrammatically in Fig. 3.

![Figure 3: Diagram of hair cell design.](image-url)
The rates of flow between the reservoirs are fixed except for the rate of release from the free pool into the cleft which is controlled by the intensity of the acoustic stimulation. In the real cell this rate of release is controlled by a permeable membrane. The membranes permeability is a function of the input stimulus intensity.

Some transmitter is lost from the cleft (and from the system) by diffusion from the cleft. This would lead to a gradual run-down of the system if it were not for a gradual replenishment of the transmitter from a 'factory' within the cell that slowly replaces lost transmitter. The amount of transmitter produced by the 'factory' is a function of how empty the free transmitter pool is.

Equations 1-4 give a complete mathematical account of the model. The flow parameters \((y, t, r, k, x)\) are constant; \(k\) is a variable function of \(s(t)\) the instantaneous amplitude of the acoustic stimulus. The \(k(t)\) equation represents a saturation function with a maximum value of \(g \cdot dt\). We can also see that with no stimulus, \(s(t)=0\), we get a non-zero value of \(k(t)\). This non-zero value is determined by parameters \(A\) and \(B\). This gives rise to spontaneous activity. We can see the response of the saturation function in Figure 4.

![Figure 4: Saturation function](image)

There is also a half-wave rectification condition in the this function. This corresponds to the fact that the cell only responds when its stereocilia (projecting hairs) move in one direction (half an oscillation). There is an adaption process which results in the steady state operation of the cell having a very low dynamic range. This means that the cell is not concerned with the intensity of the input waveform but with the shape of it. This effect is caused by depletion of the free transmitter pool in response to stimulation because the rate of release is greater than the rate of replenishment from the reprocessing store and the 'factory'. Sudden stimulation results in an initially large cleft content value which quickly decays to a steady state value.

4 Hair cell implementation

4.1 Implementation route

Our hair cell was initially modelled in behavioural VHDL. This model was used as a reference to verify the following design stages. The code was then converted to synthesizable VHDL code. Once the synthesized netlist had been verified we down-loaded the design onto an FPGA. We tested the hardware design using PC based test-bench software. The testbenches at each design stage accepted the same format of stimulus file. This means that we could apply an identical stimulus file to all stages of the design to verify their behaviour.

4.2 Cell design structure

The design can be split into 3 top-level blocks (Figure 5). The REG_SET block latches the input, produces the output at the end of each calculation, and stores the previous values of the internal reservoir levels \((q(t), c(t), w(t))\).

CALC_FLOW calculates how much flow of transmitter substance there is between the three reservoirs. This block contains one divider, six multipliers, and a subtracter. The multipliers are relatively small - they are unsigned and multiply by a constant. The divider is necessary to implement the saturation function and is the most complex arithmetic unit in our design.

CALC_RES takes the previous reservoir levels and the flow values to calculate the new reservoir levels. This is simply a process of adding and
subtracting the flow values to and from the old reservoir values.

4.3 Arithmetic and resolution issues

Reservoir levels and flow values are represented in 16 bit unsigned format. All 16 bits are fractional to represent the range 0 to 1. This is the normalised range of values where 1 is the maximum amount of transmitter possible in the free transmitter pool. 'Ripple carry' adders and subtracters and 'shift-and-add' multipliers are used. We also use a 'conditional shift-and-subtract' divider. Internal variables are updated and a new result is produced every 50 μs (epoch time).

5 Test results and analysis

5.1 Response to pure tone stimulus

Figure 6(A) shows the input to the hair cell model. It is a 1 kHz tone which starts at intensity 0 dB and then steps up to 50 dB and finally 70 dB. Figure 6(B) shows the response of the floating-point implementation of the model and Figure 6(c) shows the response of our 16 bit implementation.

The response of the cell can be split into the onset response and the steady-state response. At onset the cell has a wide dynamic range, yet in the steady state it has a small dynamic range. In other words the onset characteristics are sensitive to a wide range of input intensities, whereas, the steady-state response remains constant above a certain low input intensity. We can see that the adaption process causes a decay in the response of the cell from the onset response to the steady state response. This performance is very close to that of the hair cell model in [2].

5.2 Hardware details

Table 1 shows the gate count and maximum clock frequency of each block and of the whole cell.

<table>
<thead>
<tr>
<th>Block</th>
<th>Size (Gates)</th>
<th>Speed (MHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>REG_SET</td>
<td>314</td>
<td>11.11</td>
</tr>
<tr>
<td>CALC_RES</td>
<td>1161</td>
<td>2.86</td>
</tr>
<tr>
<td>CALC_FLOW</td>
<td>1794</td>
<td>1.68</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td>3269</td>
<td>1.68</td>
</tr>
</tbody>
</table>

Table 1: Hair Cell Complexity and Speed
We can see that the REG_SET block is the fastest and least complex block. This is due to it having the least combinational logic. CALC_RES has only adders and registers. This gives it a simple structure. CALC_FLOW is more complex with its multipliers and divider. This gives us more signals to monitor during verification. The cell was implemented in a XILINX XC4013 device.

It takes 42 clock cycles for each iteration. A maximum clock frequency of 1.68 MHz gives a new result in 25.02 μs.

6 Conclusions

We have presented a hardware implementation of the mammalian inner hair cell. While the design of the cell is relatively primitive it does represent the first digital implementation of the inner hair cell. It is also the most representative in terms of the cell's internal operation. Further work on more efficient designs is underway. We hope also to implement the neuromorphic pitch detection system in real-time digital hardware. Again we will target FPGA technology.

References


VHDL-based Design of Biologically Inspired Pitch Detection System

S.C.Lim, A.R. Temple, S. Jones, and R. Meddis
Department of Electronic and Electrical Engineering, Loughborough University, Loughborough, Leics., LE11 3TU, United Kingdom.
Email: s.r.jones@lboro.ac.uk

1Department of Psychology
University of Essex
United Kingdom

Abstract

Engineering researchers at Loughborough University and hearing researchers at Essex University are working together to produce a real-time digital electronic pitch extraction system based on neuromorphic principles and implemented in digital technology. This paper reports on the VHDL simulations of the system which demonstrates good and robust pitch tracking across the human voice range.

1. Introduction

The capability of living creatures to perform sophisticated signal processing is well-known. As the capacity of microelectronics technology improves and our understanding of the cellular structure of neural tissue develops, it becomes increasingly feasible to address the design of neuromorphic circuitry to provide electronic equivalents of the signal processing found in living creatures.

Vision, with its high data rates and 2-dimensional input signals, currently remains outside the capability of digital technologies. However, auditory systems, with their 1-dimensional representation and lower sampling rates appear well-suited to the multi-million transistor capability of IC technology. Furthermore, there are a number of distinct application areas where such systems can be of benefit. Firstly, by acting as a substitute for living creatures they can reduce the need for experimentation. Secondly, for people with severe damage to the auditory brainstem they can act as substitute "ears" with a greater capability than cochlear implants. More speculatively, it appears from studies made, that substantial pre-processing of data (e.g. the recognition of plosives and pitch detection) occurs in the initial part of the mammalian auditory brainstem. By utilizing such a system, it may be possible to develop more advanced speech recognition systems which operate with the rich and complex representation of the sound source provided by the early stages of the brainstem.

In following sections, we discuss the goal of the work and explain the theory on which we base our models. The need to simplify this theory and the simplifications made are explained in section 3. Graphical results from each stage of our simulations are shown in section 4 together with an analysis of these results.

2. The four stage pitch system

2.1. Aim of this system

The goal of this work is to implement a system that emulates the process of pitch extraction in the mammalian auditory brainstem in real time. The input is a signal of range 80 - 5 kHz. This system identifies the pitch (fundamental frequency) of this complex input signal, within the range 80 - 300 Hz (equivalent to the pitch range of human speech) to a resolution of 20 logarithmically spaced channels. The output of these channels will be displayed on a monitor in real-time.
This operation is believed to be undertaken in the initial part of the mammalian auditory brainstem. By implementing such a system, we will be able to verify the feasibility of this conjecture and experiment with the different components’ uses and explore their role.

2.2. System principles

The system comprises 4 stages. The biological basis for such a structure can be found in [1]. The system models the different processes from the auditory periphery, to the cells in the Inferior Colliculus (IC) nucleus of the auditory brainstem. A brief explanation of this process follows.

- The first stage of the model corresponds to the cochlear mechanical filtering. This is realized in our system through a bank of bandpass filters. Each filter extracts a separate bandlimited part of the input signal and has a bandwidth of 10% of its center frequency. If the input to the system is a complex or natural sound, the output from each filter resembles an amplitude modulated sine wave. This means we can initially test the system using a simple modulated signal without requiring human speech as input.

- The signals from the cochlear filtering are then fed into the second stage that corresponds to the inner hair cells within the cochlea. These cells convert the amplitude of the sound into a series of spikes which are transmitted along the auditory nerve (AN). These spikes take the form of very rapid and brief increases in the voltage potential being transmitted. The hair cell only produces these spikes during the positive cycle of the input stimulus. The spike rate is a function of the stimulus intensity and is also subject to saturation and adaption. Adaption is an effect which appears directly after the input intensity to the cell changes: the cells’ output intensity initially changes according to input intensity change but then decays, over time, to an equilibrium intensity.

- Each hair cell output is then fed into the third stage comprising an array of stellate chopper cells. These cells are predominant in the cochlear nucleus. When stimulated with an amplitude modulated signal, it is found that these cells have a phase locking capability and will amplify signal modulation at a specific Best Modulation Frequency (BMF). The BMF of a stellate chopper cell is the stimulus modulation frequency with which the cell shows greatest phase locking. The stellate cells in an array are arranged in banks. Each cell in a bank has the same BMF while different banks have different BMFs. When the input to a bank of stellate cells has a modulation frequency close to the cells' BMF, these cells will all generate spikes at approximately the same time.

- The output from each bank of stellate chopper cells is fed into a coincidence cell. This type of cell is believed to reside in the Inferior Colliculus [1]. Each coincidence cell spikes when a large number of its input stellate chopper cells all spike simultaneously. Based on the spike rate of the coincidence cell associated with each bank of stellate cells, it is possible to determine the pitch of the stimulus.

---

Figure 1. A simplified Diagram of the four stage model for pitch extraction.
This system effectively converts the amplitude modulation information (periodotopic information) into spike rate information. The pitch of the input stimulus is signaled by the most active coincidence cell.

2.3. Why hardware implementation?

It is possible to simulate these circuits in software. However, the large number of cells involved, allied to their fine-grain asynchronous parallel operation, imposes a formidable computational load. While parallel machines ease this problem, the full-scale and real-time operations needed for experimentation are as yet beyond the capacity of simulation based approaches. In contrast a hardware implementation offers a low-cost and high-speed capability that would appear to be an attractive approach.

3. Design details

3.1. Abstractions from biology

3.1.1. Cochlear filters. The cochlear filters in the auditory periphery separate the input stimulus into a number of frequency bands which are then processed separately in the auditory brainstem. We implement these cochlear filters using first-order Butterworth band-pass filters. This provides a sufficiently sharp and smooth filter response with the advantage of being simple in design. In our design there are 30 cochlear channels filters each with center frequencies ranging from 80 Hz to 105 kHz in logarithmic steps and bandwidths 10% of their center frequency. This range is where the significant frequency components are in human speech. Since each cochlear channel is identical after this initial filtering process, we will, therefore, consider the design of a single channel from now on.

3.1.2. Hair cell. The original Meddis hair cell model [2,3] generates spikes in the auditory nerve fibers. The model generates the spiking function using multiple random numbers together with a calculated spiking probability value. These spikes are then transduced into a current when they arrive at the stellate cell dendrites. We simplified this process by adding noise to the hair cell spiking probability value then applying the result of this to the stellate cell dendrites directly. A single Linear Feedback Shift Register (LFSR) is used to create all of the pseudo-random numbers required in this implementation. A quadratic function uses these pseudo-random numbers to apply noise to the hair cell probability such that the noise component tends to zero if the hair cell probability tends towards 0 or 1. We use our artificial synapse to produce 40 different noise sources for the single hair cell signal.

3.1.3. Auditory nerve fibers. The multiple AN fiber signals are processed by the stellate cell dendrite filters. These dendrites transmit the AN fiber signals to the stellate chopper cell nucleus. They also have low-pass characteristics. For this reason we use first-order Butterworth filters to represent the low-pass dendrite filters. Each of our 40 noisy signals is filtered by a stellate cell dendrite filter.

3.1.4. Stellate chopper cells. Stellate chopper cells spike at a fixed rate for a fixed amplitude input. A Fixed spiking rate represents an average inter-spike period, however, two consecutive inter-spike periods are quite likely to be different considering the noise injected into the system. A constant inter-spike period (phase locking) only occurs when the modulation frequency of the input stimulus matches the BMF of the stellate cell. Our stellate chopper cell model [4] is a version of the model adapted by Hewitt et al [5] from MacGregors cell model [6]. The stellate chopper cell array has dimensions of 20 rows by 40 columns. Each stellate chopper cell in one row has the same BMF. A row of cells in our model is equivalent to the bank of cells in the real

Figure 2. Single cochlear channel architecture.
neural system. The BMFs of each row vary from 80 Hz to 300 Hz in logarithmic steps. Each of the 40 different noisy filtered signals that come from the dendrites are fed into a column of stellate chopper cells. Therefore, each of the cells in a stellate chopper cell row receives a different input. If the modulation frequency of the input signal matches the BMF of a particular row then the stellate choppers cells will spike at a similar time (high synchronization). If not then the separate noise sources applied to each cell in a row results in their synchronization degrading with time.

3.1.5. Coincidence cells. The coincidence cell is sensitive to how many stellate cells in one bank are spiking within a certain period (epoch). The epoch time for each coincidence cell varies depending on the BMF of its stellate bank. More specifically it is proportional to the period of the BMF of the stellate cell row associated with that coincidence cell. The input spikes are transduced into a current in a similar fashion to dendrites. The cell itself produces a single spike for constant stimulus instead of the regular series of spikes associated with the stellate cell. In our design there are 20 coincidence cells in a cochlear channel, one for each of the stellate chopper cell bank (row). If the number of stellate chopper cells spiking during any one epoch exceeds a preset threshold then the coincidence cell produces a single spike. Our coincidence cell design simply adds the spiking values up over a set period. The most active coincidence cell in the coincidence cell bank tells us the modulation frequency and thus the pitch of the input signal.

3.2. Architecture

Figure 2 shows the implementation of a single channel. Note the number of instantiations of each block is shown. All data is stored in 16 bit format. Our previous 13 bit implementation of the stellate cell [4] provides the minimum accuracy to maintain functionality of the cell. However, a 16 bit implementation is chosen for this system to ensure that we can, first and foremost, produce an accurate system rather than a highly optimized one. In a biological system spiking is used for communication. We represent the spiking of an output by a single-bit binary variable. The sample rate of the system is 20 kHz. This gives us sufficient sampling for the 5 kHz bandwidth input signal. In this, the first step, only a single cochlear channel is implemented.
3.3 VHDL implementation

VHDL is a concurrent hardware description language. Its concurrency allows us to implement highly parallel, logic based designs easily. Since it is a hardware description language it is possible, through the use of synthesis tools, to generate circuits from the VHDL code. It also enables us to model the system operation and to progressively develop the system design.

The operation of our VHDL system model is cross-referenced against the software model in [7]. Initially we start with a behavioral description of the system. Each module is implemented using logic based functional and arithmetic modules. This allows us to design a system that will match the logical operations and data flow of the eventual hardware design. This behavioral design has been implemented and simulated.

4. Results and analysis

4.1 System calibration and test

The system is calibrated by first adjusting Gain\textsubscript{1} (Figure 2) such that when there is no stimulus applied to the system there is still activity in the stellate chopper cells. This activity is a direct result of the spontaneous activity inherent in the hair cell. Gain\textsubscript{1} is then adjusted to vary the input stimulus level. We describe the 0 dB level as the intensity of input stimulus which results in a noticeable increase in activity within the stellate cell array. We use a coincidence threshold of 8 for all our coincidence cells. This means that if 8 or more stellate cells in a row spike simultaneously, the coincidence cell will spike. To test our design we use stimulus signals with carrier frequency 3333 Hz, modulation frequencies ranging from 80 Hz to 300 Hz. The modulation index is 35%. This carrier frequency is chosen to match the center frequency of the cochlear filter.

4.2 Simulation results

Figure 3 shows each of the signal processing stages within the system. We can see that the stimulus is virtually unchanged as it passes through the cochlear filter. The half-wave rectification characteristics of the hair cell can be seen. The saturation of the hair cell is seen in the slight loss of modulation of the stimulus. Adaptation is not demonstrated here as the stimulus intensity level does not change after being switched on.

We can see how the added noise at the synapse disrupts the hair cell output. The amount of noise added is not, however, so much as to destroy the modulation information after dendrite filtering. The synapse and dendrite graphs shown here are each one from a set of 40. We show here the response of a single stellate chopper cell from the 20 by 40 array. The sharp peaks in the response curve show when the cell spikes.

Figure 4 demonstrates the output from the coincidence bank for 3 signals with different modulation frequencies. The results show that when the modulation rate of the applied stimulus matches the BMF of a particular stellate bank, the corresponding coincidence cell has the highest output spike rate. We can also see the difference in this activity compared to the other coincidence cells. As can be seen clear recognition of the pitch is achieved. The performance corresponds closely to that observed in living creatures [1].
5. Current status

At the time of writing we have implemented a single channel of the pitch detection system in synthesizable VHDL code. This implementation contains a large number of parameters which define the behavior of the system. Once we have a complete hardware implementation of the single channel we hope to carry out a number of behavioral experiments on it by varying the system parameters. This will help us characterize the system in more detail.

Implementation of the system in hardware has, in fact, begun. With a single FPGA we can only implement one component of the design at a time. To date we have implemented and successfully verified each of the components of the model. As we have discussed, these components are: a cochlear (band-pass) filter, a hair cell, a synapse, a dendrite filter bank, a stellate cell row, and a coincidence cell. We have used bit serial algorithms to implement the arithmetic building blocks required. This provides us with the speed to run the above components in real time while allowing them to individually fit within a single FPGA.

We now plan to verify communication between the components by using the output from one as the input to the next. When we verify a component in hardware we provide a stimulus file for the test-bench software. This in turn produces an output file containing the hardware response to the stimulus. This output file can easily be reformatted into a stimulus file for the next component in the design. Using the test-bench software to check inter-component communication will ensure that the system works as a whole.

Once the communication between the components has been verified we then hope to implement the whole channel on an array of FPGAs.

6. References

[7] L.P. o'Mard, M.J. Hewitt, R. Meddis. 'LUTEar Core Routine Library Ver1.5.2,' Speech and Hearing Laboratory, Loughborough University, 1996.
Digital Hardware Implementation of a Neuromorphic Pitch Extraction System.

Digital Hardware Implementation of a Neuromorphic Pitch Extraction System.

S.C. Lin, A.R. Temple and S.R. Jones
Department of Electronic and Electrical Engineering
Loughborough University, Loughborough,
Leics. LE11 3TU, United Kingdom.
Email: S.R.Jones@lboro.ac.uk

1. Introduction

There is clear evidence of the mammalian brain as an outstanding signal processing device, undertaking image and auditory signal processing substantially beyond the capability of algorithmic techniques. The rapid developments of semiconductor technology has meant that the complexity of the systems we can design has increased rapidly over the last 20 years and is likely to continue to do so in the next 10. Consequently, it is timely to consider whether we can use advanced microelectronics technology to implement the processing found in living brains.

Attempting to implement the cortex is problematical as a clear understanding of the operation and purpose of the cells in the cortex as yet eludes us. However, the visual and auditory brainstems are easier to model as (a) we can directly control the input these parts receive and (b) a clear knowledge of the structure, function and purpose of the cells in the early part of the brainstem is known through extensive physiological modelling.

The long-term goal of our work is to understand how best to construct microelectronic implementations of the brainstem, which capture the key behaviour of these cells and yet map efficiently into digital technology. The auditory brainstem is probably the best one to study first as it is both simpler and processes less data than the visual brainstem, yet still has useful applications in music and speech processing.

The first work we have undertaken reports the implementation of the signal processing found in the early part of the mammalian auditory brainstem. In particular we focus on a known phenomena that occurs in the early stages: the detection of fundamental pitch within a complex sound.

The system has been designed and fully simulated in VHDL and shown to be able to detect fundamental pitch within the human voice range (80Hz-300Hz); key cell components have been implemented and tested in FPGA technology and shown to have good correspondence with physiological measurements.

Overall, the work while in the early stages demonstrates two of the key goals of our work (1) the practical utility of this approach and (2) the capacity of digital designs to accurately capture physiological behaviour.

2. Background

2.1 The Auditory System.

The hearing system can first be viewed as two parts, the auditory periphery and the auditory brainstem. The auditory periphery performs signal processing through the use of mechanical structures, while the brainstem uses neural circuits.

The auditory periphery is subdivided into three areas, the outer, middle and inner ear. The outer and middle ear serve as a physical structure to selectively collect and filter the sound. This sound then causes vibrations on the ear-drum (tympanic membrane). The middle ear bone structures then transfer the acoustical vibration on the ear drum into the inner ear. This provides the necessary acoustical matching between the air and the fluid of the cochlear in inner ear. Contained in the cochlear is the
bassilar membrane. When stimulated by sound, this membrane vibrates with a traveling wave type motion which peaks at locations along membrane relative to the frequency of the sound. Inner hair cells along the membrane then converts the mechanical vibrations into neural impulses which are then transferred to the auditory brainstem. These inner hair cell are the beginnings of the auditory brainstem.

![Figure 1: Structure of the ear](image)

![Figure 2: A Simplified Diagram of Brainstem nuclei structure and circuits.](image)

The auditory brainstem is made up of a large number of different types of neural cell. Cells of the same type are grouped together in areas of the brainstem. Figure 2 shows a simplified organisation of these groups. The discovery of neural circuits has led researchers to believe that preprocessing of the auditory stimulus is performed in the auditory brainstem. An example of this is the localisation of sound which is achieved by neural circuits that are sensitive to interaural time difference [1] and intensity differences [2].

2.2 The Four Layer Pitch Extraction System

![Figure 3: The Four Stage Model for pitch detection.](image)

to their response to the modulation frequency, or periodicity, of the input stimulus. Since all complex and natural sounds have in common their repetition rate of amplitude fluctuation, it was suggested
One theory of pitch preprocessing in the auditory brainstem is the four stage model for pitch detection [3]. The four stage model is shown in figure 3. This models the pitch system from the auditory periphery to the Inferior Colliculus (IC) of the brainstem.

The Cochlear mechanical filtering forms the first stage of the system. This separates the input stimulus into frequency bands. This stage is modeled by a bank of bandpass filters with different center frequencies.

The hair cells form the next stage of the system. Each cell converts the sound intensity and periodic information from its cochlear filter into neural spikes on the Auditory Nerve (AN). These spikes are generated by a transmitter substance which produces during the positive cycle of the haircell's input signal. The amount of transmitter substance produced is related to the intensity of the input signal as well as the saturation and adaptation characteristics of the cell. Adaption is a process whereby production of transmitter substance increases or decreases according to a step increase or decrease in signal intensity before decaying over time to a steady state average rate.

The third stage is made up of an array of stellate chopper cells. These cells form a major part of the cochlear nucleus. The spikes from the hair cell travel along the AN and through the dendrite filter at the input of each stellate chopper cell. For a filtered input of constant amplitude, the cell will produce a neural pulse train at a relatively constant rate. From physiological studies [4] and software modeling [5], it is known that these cells are able to amplify amplitude modulations. This is because of the ability of these cells to phase lock to the modulation rate when the rate of modulation is close to the firing rate of the stellate cell. Most natural sounds, especially speech, can be broken down into a number of modulated pure tones with a modulation rate equal to the base pitch of the original sound. Therefore, the ability of stellate chopper cells to phase lock to the modulation rate will be critical to the system operation. More details can be found in [3].

Each hair cell output will, therefore, feed multiple banks of chopper cells. Each bank has a different Best Modulating Frequency (BMF) to detect a range of pitch frequencies.

The last layer is made up of coincidence cells. These model the cells found in the Inferior Colliculus. Each of these cells receives neural spikes from a bank of stellate cells. When a large number of stellate chopper cells in a bank spike at the same time, the coincidence cell responds by generating a spike. Hence, if the input has a pitch or modulating rate near one of the BMFs, then all the coincidence cells that are connected to banks having that BMF will have high output spike activity. This can be used to determine the pitch of the input sound.

This system for pitch detection has been modeled in software [3]. However, since a large number of cells are needed to construct a working system, software simulations are very slow. Here, a hardware implementation can provide a way to realise the system in real time.

3. Abstraction from biology and cell implementations

Before the system can be implemented, work has to be done to ensure that all cells and processing units can be modeled in hardware. These models have to provide functionality similar to the biological ones but with enough simplification so as to optimise the final implementations. These models are then first implemented and simulated in synthesisable VHDL before being implemented using FPGAs for verification of hardware operation. We use the Xilinx 4013-6 FPGA for our verifications of our designs. All the units are presently implemented using the bit-serial with a data represented using 16 bits. This bit-width is chosen to provide the necessary resolution needed by the hair cell for stability. Space does not permit a detailed explanation of all the cell developed. Therefore we only provide details of 2 key components and deal later with the complete system operation. The 2 key components are the hair cell and the stellate cell.
There exists many probabilistic models of the hair cell. But a model must be able to model the saturation, adaptation, onset and halfwave rectification behaviour of the actual hair cell. We chose to representations of a hair cell, but is also computationally efficient. This model describes the cellular processes by calculating transmitter flow among reservoirs within the cell and the synaptic cleft as a amount in the synapse, the output of the cell, is then used as a probability value to determine the spiking of the auditory fibers connected to it.

then into an FPGA. The structure of a single hair cell design is shown in Figure 4. This implementation has three top level units, REG_SET, CALC_FLOW and CALC_RES. REG_SET is substance within the cell and CALC_RES calculate the reservoir levels. The cell operation clock rate, required clock cycles per epoch and implementation size is shown in table 1. Its speed is well

The behaviour of the implementation shown in figure 5(b) is similar to that by Meddis [6] Figure 5(c). The response of the model demonstrates the characteristics of the inner hair cell. The onset response a decay in the output of the cell down towards a steady-state response.
3.2 The stellate cell dendrite unit.

The stellate cell can be divided into two parts, the dendrite and the cell soma. The dendrite can be modeled using a digital IIR first order low pass Butterworth filter. The cutoff frequency used is 300Hz [5]. The size and speed information of this filter is given in Table 1.

3.3 The stellate cell soma unit.

For the cell soma, we chose to base our design on MacGregor’s point-neuron10 cell soma model [8]. The MacGregor cell model represents the spiking process using differential equations that describes the interaction between the cell membrane potential, the potassium conductance and the threshold potential. Since, for a stellate chopper cell, the threshold stays relatively constant, we simplified the design by fixing the threshold. Figure 6 shows the implemented structure of the unit. Table 1 detailed its size, clock speed and required clock rate for one epoch. Its speed allows 40 cell units to be multiplexed using a single implementation in real-time. Figure 7 show the functional results of the cells.

![Image](image-url)

Figure 6: Implemented structure of the stellate cell soma unit.

![Image](image-url)

Figure 7: Circuit response (a) and actual neural response (b) to 20mV(0.6nA with 33mΩ input impedance) depolarising and hyperpolarizing potential. [(b) redrawn from [9], Figure 5]

<table>
<thead>
<tr>
<th>Processing unit</th>
<th>Implementation Size</th>
<th>Clock speed (MHz)</th>
<th>clock cycles per epoch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hair Cell</td>
<td>3269</td>
<td>1.68</td>
<td>42</td>
</tr>
<tr>
<td>Dendrite Filter</td>
<td>747</td>
<td>23.6</td>
<td>32</td>
</tr>
<tr>
<td>Stellate Cell Soma</td>
<td>1780</td>
<td>32.1</td>
<td>32</td>
</tr>
</tbody>
</table>

Table 1: Implementation Size, Clock Speed and Clock Cycles per Epoch.
4. System implementation

4.1 Structure of abstracted system

For experimentation, we chose first to implement a single cochlear channel. The structure of one channel of our system can be seen in Figure 8. To allow our system to detect 20 pitch frequencies in the range 80 Hz to 300 Hz in logarithmic steps 20 groups of stellate chopper cells are needed. Each group's BMF matches one of the pitch frequencies we are trying to detect. These groups are called rows. Each row contains 40 stellate chopper cells of the same BMF. To simulate the stochastic nature of the auditory nerve fibres we use a random number generator to add noise to the output of the hair cell. Each stellate cell in a row needs an independent noise source to provide an uncorrelated noise to each cell. However, each noise source can be shared among a column of stellate cells. This allows one noise generator and dendrite filter to be used for each column of 20 stellate cell somas. This results in a total of 40 noise and dendrite filter units. Finally, 20 coincidence cells are used, one for each row of stellate chopper cells. These just count the number of spikes that occur in a single epoch and produce a spike when the number exceeds a threshold value. The BMF associated with the most active coincidence cell is the pitch of the input signal to the system.

Figure 8: structure of a single cochlear channel of the 4 layer pitch detection circuit.

4.2 VHDL modeling of the system

The system has been modeled in synthesizable VHDL using 16 bit data representation. The spikes produced by the stellate chopper cells and the coincidence cells, however, use single bit representation. The input sound sample rate of the system is 20 kHz gives us a sample period (epoch time) of 50 μs. This gives us sufficient bandwidth of 5 kHz to cover most of speech frequency spectrum. Multiplexing is use extensively in the system so that only one synaptic noise unit, one dendrite filter and 20 stellate cell somas are needed to emulate all their respective types.

4.3 VHDL simulation and results

Before testing the behaviour of the system we have to calibrate it. The first stage of calibration is to ensure that the stellate chopper cells spike in the absence of a stimulus to the system. This is to emulate spontaneous activity inherent in the biological system that originates from the haircell's spontaneous output. We adjust Gain₂ (see Figure 8) to obtain a spontaneous spike rate of about 20 spikes/s for each cell. Gain₁ is used to control input signal intensity. We then define the 0 dB point of the system as the input stimulus intensity which results in a noticeable increase in the spiking rate of the stellate chopper cells.

To test the behaviour of the single cochlear channel we apply a signal with carrier frequency 3333 Hz, and modulation frequencies ranging from 80 Hz to 300 Hz. The modulation index is 35% and the
signal intensity is 30 dB. We have also set the coincidence cell threshold at 8. The centre frequency of the cochlear filter is set to match the carrier frequency of the input signal so as to compensate for the fact that only one cochlear channel is implemented.

The individual response of the system components are shown in Figure 9. We can clearly see the half-wave rectification characteristics of the hair cell together with its compression effect. This compression is demonstrated by the slight loss of modulation information.

The slight disruption of the hair cell output caused by the synaptic noise injection can also be seen. It is important for the modulation information not to be destroyed by too much noise. The dendritic filtering shows that it has extracted the modulation envelope. Here the natural frequency of the stellate cell matches the modulation frequency of the signal and therefore the cell spikes will become in phase with the peak of the input signal.

Figure 9: Response of each pitch system's component.

The response of the coincidence cell bank to 3 signals of different modulation frequency can be seen in figure 10. The results show that when the modulation frequency of the input stimulus matches the BMF of a stellate chopper cell row, the associated coincidence cell produces a high spiking rate. This response corresponds to a clear identification of pitch.

Figure 10: Response of the coincidence cells to a number of stimuli of different modulation frequencies.
6 Future Work

The next phase is to implement the single channel onto an array of FPGAs. Once this has been achieved we have to find more efficient ways of implementing the system. One area of development could be in reducing the sample rate of the stellate chopper cells. We can see that after dendritic filtering these cells are subject to a stimulus of bandwidth 300 Hz. This means that a sample rate of 20 kHz is somewhat excessive. We also plan to experiment with different number systems and arithmetic techniques.

8. Acknowledgments

We would like to thank all members of the Electronic Systems Design Group at Loughborough University for their assistance.

9. References.


Digital Implementation of a Stellate Cell Using the MacGregor Cell Model

A.R. Temple, S.C. Lim, S.R. Jones
Electronic Systems Design Group
Department of Electronic & Electrical Engineering
Loughborough University
Loughborough, United Kingdom. LE11 3TU

Abstract

This paper introduces the digital implementation of an auditory stellate cell based on the MacGregor cellular model. This cell is one of the basic components of the mammalian auditory system. The method of implementation is described and the digital implementation is tested and compared with the physiological data. Results show that the digital implementation has good correspondence with the actual stellate cells.

1. Introduction

1.1 Motivation

Over the past decade there has been substantial research into the use of neural networks and neuromorphic circuitry for emulating the signal processing in the visual components of the brainstem (e.g. pattern recognition and general machine vision). There has in contrast hardly been any interest shown in the use of neuromorphic circuitry for signal processing in the auditory brainstem. Indeed, to our knowledge only ourselves and van Schaik at Lausanne [1, 2] are working in this area today. Van Schaik's work focuses on analogue implementation and our own work on digital realisations.

Research in the electronic implementation of the signal processing within the mammalian brainstem is useful for a number of reasons including:

- There is potential benefit in the long-term in application areas of speech recognition. Much of speech processing (e.g. recognising plosives such as 'p' and 'k') occurs in the initial part of the brainstem.
- Use in advanced hearing aids for people with damage beyond that repairable with current cochlear implants.
- The availability of simulations and real implementations of components that constitute the auditory sub-system can be used to gain a clearer insight into the operation of such systems.
- Software models are based on sequential-code or coarse-grain parallel implementations, which only capture imperfectly the dynamics of the fine-grain asynchronous concurrency found in living systems. In contrast, electronic simulations and implementations can model much more precisely such behaviour, enabling a clearer insight into the operation of the auditory subsystem to be made.
- The access to more accurate real-time cell-based models may also contribute to a reduction in our need to experiment on living creatures.

1.2 Goal of work

The long-term goal of the work is to understand how to design efficient and effective digital implementations of the signal processing within the auditory brainstem. We can also use these models to...
understand and explore the actual operation of the cells and collections of different cells in real time.

Our first design project aims to produce a pitch extraction system based on neurological principles. This system can take as input a sound signal and resolve the pitch of the signal with an accuracy of 10Hz over a 80Hz-300Hz range (this corresponds to the pitch of the human voice). We are using multiple FPGAs as our implementation technology and VHDL and logic synthesis as the design tools.

The first stage of this work is to construct the component cells found in the auditory brainstem. In this paper we report on the design of one of the cells 'the stellate cell'. An array of several thousand stellate cells is one of the early major components in the auditory brainstem. The purpose of the stellate cell is described later on in this paper.

1.3 Structure of paper

In Section 2 we give an overview of the signal processing that occurs within the auditory brainstem, starting from the external ear itself through to the neural nuclei within the auditory brainstem. The MacGregor cell model which we are using as a generalised building block within our design is detailed in Section 3. We describe the model and motivation for using it. Sections 4 and 5 discuss implementation issues, including arithmetic and data representation approaches together with a consideration of architectural issues. In Section 6 we report experimental results from the first digital implementation of the stellate cell and compare hardware design performance with biological data.

2. Signal Processing In The Auditory System

Initially, we can think of the auditory system as being split into the auditory periphery and the auditory brainstem. The auditory periphery is the mechanical 'front end' to the auditory brainstem. The auditory brainstem is the neural section.

2.1 Auditory periphery

The auditory periphery comprises the outer, middle, and inner ear. The outer ear comprises the pinna, concha, and ear canal. These features act as filters to the acoustic signals entering the ear.

![Auditory periphery](image)

Figure 1: Auditory periphery [3].

The middle ear is made up of the tympanic membrane and three ossicles. These ossicles are attached to the cochlea via the oval window. The cochlea contains the hair cells which are connected to the auditory nerve (AN) fibres. These inner hair cells are fundamental to the electronic implementation of the auditory pathway as these cells are responsible for converting mechanical vibrations into electrical impulses.

![Neural cell structure](image)

Figure 2: Neural cell structure.
2.2 Auditory brainstem

The auditory brainstem receives its input from the auditory nerve. As mentioned in the previous section, the AN is stimulated by inner hair cells which reside in the cochlea. Each AN fibre is finely tuned to a certain frequency band. The auditory brainstem comprises a number of nuclei, each of which contain different types of cells and perform different functions.

It is not yet fully understood what exactly these nuclei do. We are aware of certain systems within the brainstem, one of which is the pitch extraction system. This system extracts the fundamental frequency from speech which is made up from a large range of frequency components.

The most predominant neural cell in this system is the stellate chopper cell which resides in the cochlear nucleus. This cell consists of the cell soma (nucleus) and dendrite fibers at the input of the soma, and the output is transmitted along the axon (see Figure 2). Each stellate cell has a best modulation frequency (BMF). The input to each stellate cell can have a random (stochastic) waveform or a noisy half-wave rectified sinusoid waveform depending on the behaviour of cells feeding its input. When a stellate cell receives input in the form of a noisy half-wave rectified sinusoid at its BMF then it produces an enhanced (less-noisy) version of this input. In other words the stellate cell has the task of enhancing phase-locking at a certain frequency.

3. The MacGregor Cell

The MacGregor cell model is based on the model by Hodgkin and Huxley [4,5,6] and is able to emulate the dynamic behaviour of neural cells. The MacGregor model is also a generic model, which means that the behaviour of the model can be made to simulate the behaviour of many different types of cells by adjusting the value of its parameters. Hence basing our design on this model provides us with both accuracy and the possibility of using the same design for many different cell types.

The cell model used here is based on MacGregor’s ‘Point Neuron’ model [7] and also on the slightly modified version used by Hewitt et al [8,9] in their simulations. We also based our parameters on the simulations by Hewitt et al. The model behaviour is described by four main differential equations. They are:

\[
\frac{dE}{dt} = \frac{-E + (V + G_k(E_k - E))}{T_{mem}}
\]

\[
\frac{dG_k}{dt} = \frac{-G_k + (b.S)}{T_{gk}}
\]

\[
\frac{dTh}{dt} = \frac{-(Th - Th_0) + c.E}{T_{th}}
\]

\[
S = \begin{cases} 1 & \text{if } E(t) > Th(t) \\ 0 & \text{otherwise} \end{cases}
\]

The first equation describes the behaviour of the membrane potential, \(E\), in relation to the input potential, \(V\), and the delay potassium conductance, \(G_k\). \(E_k\) is the cell’s resting potential against which \(E\) is measured. \(T_{mem}\) is the time constant of the cell which determines the rate of change of the membrane voltage. The input potential, \(V\), has an incremental effect on the value of \(E\) whereas \(G_k\) has a decremental effect on \(E\).

Equation 2 deals with the behaviour of the delay potassium conductance, \(G_k\), and is responsible for the refractory (resting) period of the neural cell. \(G_k\) decays exponentially in time towards zero. When the cell spikes, however, there is an instantaneous increase in the value of \(G_k\). \(T_{gk}\) determines the rate of decay of the potassium conductance.

Equation 3 calculates the threshold, \(Th\), of the cell. Here, \(Th_0\) is the resting threshold, \(T_{th}\) is the time constant that determines the rate of change and \(c\) determines the degree of change. When \(E\) has a positive value it has an incremental effect on the value of \(Th\) and when \(E\) has a negative value it has a decremental effect on the value \(Th\).
The spiking variable $S$ is described in Equation 4. This variable has value 1 when the membrane potential exceeds the threshold and 0 otherwise.

Simulation of this model in software or digital hardware can be performed by discrete integration. Since MacGregor's equations are formulated for simulation using 1ms time interval and exponential methods, the second equation needs to be slightly modified for discrete integration and a shorter step (epoch) time during simulation. Also, as soon as $E$ exceeds the threshold, $S$ is set to one and immediately reset. This means that although the membrane potential exceeds the threshold for a small but finite period, we only wish to record the fact that the event has happened. The discrete time model for the variable $Gk$ is shown in Equation 5.

$$Gk(t) = Gk(t-1) + \Delta t \left( \frac{-Gk(t+1)}{T_{mk}} \right) + B(S(t+1)) \quad \ldots \quad (5)$$

### 4. Modelling the stellate cell

The stellate cell is chosen for our first implementation due to its well-known characteristics which fit well into this model. These cells are also one of the main building blocks in large auditory systems. To model the stellate cell using the MacGregor cell model, values need to be defined for the parameters. Some parameters, like the membrane time constant $T_{mem}$, input resistance, resting potential and threshold level are specified by Oertel's data [10,11,12] while others are derived empirically to approximate the physiological results in Figure 6.

Since the threshold remains relatively constant, the model's threshold equation (Equation 3) can be omitted with little effect.

### 5 Cell Design And Implementation

#### 5.1 Cell Structure

The top level block diagram of the stellate cell implementation is shown in Figure 3. The equations in this implementation run in parallel. Each epoch cycle starts with the previous variable values and the input voltage $V_{in}(t)$ being fed to each module from a storage register. The results of each equation module are then fed back into the register to be stored for the next iteration. The 'Variable Register and Control Logic' module also provides control signals to the arithmetic modules which include clock and synchronisation signals.

![Diagram of cell implementation](image)

#### 5.2 Equation Modules

Each equation module is implemented using arithmetic units as shown in Figure 4. We can see that the transmembrane potential equation is the most complex. By virtue of its complexity, this equation is on the critical path of the circuit. This means that we need to optimise the system with respect to this part of the circuit.
5.3 Arithmetic Units

Addition is performed by a ripple carry adder. Multiplication is implemented using a shift-and-add algorithm. This comprises a single vector adder and some control logic. The system is clocked at the addition level. In other words each addition takes one clock cycle. The comparator and multiplexer are based on standard combinational logic circuit designs.

5.4 Resolution Issues

The selection of resolution is critical for cell implementation since it determines the accuracy and size of the design. Software based experiments were performed to compare resolution-limited implementations to that of a double floating point number implementation. It was found that three variables affect the accuracy of the model. They are the epoch time, dt, and the resolution of the variables Gk and E. The actual required accuracy needed for acceptable operation is not known. However, the general consensus is that accuracy in the biological system is very poor. We set a fixed dt time of 0.05 ms (to allow for a sample rate of 20kHz). Taking into account the epoch time and an arbitrary accuracy requirement of 95% we chose a standard resolution of 13 bits. This relates to 3 whole and 10 fractional bits for Gk and 8 whole and 5 fractional bits for E.

5.5 Implementation Route

The first stage is to describe the whole design in structural VHDL. After testing the VHDL code, we synthesised the code into a digital logic circuit using FPGA technology. Once the synthesised and back-annotated circuit had been tested, it was downloaded onto the FPGA. The FPGA is linked to a PC via an I/O card. We have written some test-bench software which downloads test vectors into the FPGA and reads the output to finally produce our graphs.

6. Test Results and analysis

6.1 Depolarisation and Hyperpolarisation test using step input voltage.

![Figure 5: Implemented cell Model response to depolarising and hyperpolarising voltage of 20mV (=0.6nA).](image)

- Depolarising
- Hyperpolarising

A step input voltage of 20mV, both hyperpolarising and depolarising, was applied to the input at 0ms and removed at 40ms. 20mV corresponds to an input current of 0.6nA. Hyperpolarisation increases the magnitude of the negative resting transmembrane potential, while depolarisation decreases the magnitude. The results are shown below in Figure 5. The cosmetic spikes in Figure 5 are added by the test-bench software when the transmembrane potential exceeds the threshold to make it easier to compare with the physiological data shown in Figure 6. As can be seen, the implementation results match the biological data.
6.2 Hardware Details

Table 1 shows the system gate size and how this is distributed among the circuit modules. The clock speed of the system is determined by the slowest module in the circuit. In this case the slowest module is the multiplier which has a maximum clock frequency of 3.05 MHz. The total number of clock cycles needed for one iteration is determined by the path through the "Transmembrane Potential Equation" block and the "Register & Control" block. This path takes the most clock cycles to complete. The result is a total of 34 clock cycles for one iteration which gives us an iteration period of 11.15 micro-seconds. With a sample period of 50 micro-seconds we can emulate one cell in real-time and three cells if the design is multiplexed.

7 Conclusion

This work is very much a first step. However, it does show that it is possible to implement one of the key cells in the auditory brainstem, with good correspondence to physiological data. Moreover we estimate such a design could be multiplexed to support several stellate cells simultaneously and in real-time. Furthermore, given a gate-count of 2248 gates per cell, this suggests that complex systems can be integrated on a single integrated circuit. Further work will involve improved architectures for cell implementation, the design of other cells in the auditory system and the construction of the pitch extraction hardware.

8 Acknowledgement

The authors would like to thank all members of the Electronic Systems Design Group at Loughborough University, Ray Meddis and Lowell O'Mard of the Auditory modelling group at Loughborough, with whom we work closely, for their patient help in explaining the operation of the mammalian auditory brainstem.

9 References


Oertel, D., 'Electrical characteristics of cells and neuronal circuitry in the cochlear nuclei studied with intracellular recordings from brain slices,' in 'Auditory Functions,' Eds. Edelman, G.M., Gall, W.E., Cowan, W.M. Published by Wiley.