Implementation of an analogue programmable Cellular Neural Network

van Engelen, J.A.E.P.

Award date:
1995

Eindhoven University
of Technology

Department of
Electrical Engineering

Electronic Circuit Design Group

THESIS

Implementation of an analogue programmable Cellular Neural Network

J.A.E.P. van Engelen

Coach: Dr. ir. J.A. Hegt
Supervisor: Prof. dr. ir. W.M.G. van Bokhoven
Date: February 1995

Abstract

In this report a design is presented for a CMOS implementation of an analogue programmable Cellular Neural Network (CNN). The main properties of these networks are that the cells (or neurons) are placed on a grid, are locally connected and contain some sort of degrading memory. Due to this structure these networks are particularly suitable for tasks such as image-processing and pattern-recognition.

Based on the original idea of Chua and Yang a Full-Range implementation (see [2]) of a Linear-Cloning-Template CNN on a square grid has been made. An important new feature of this design is the use of 2-quadrant multipliers and positive valued inputs and outputs for the implementation of the variable template-elements. The 2-quadrant multiplications are implemented using two-transistor multipliers, consisting of two MOS transistors in linear mode. As a result, the total implementation area will be reduced significantly.

Another aspect of this design is the direct relation between input and output of a cell, resulting in template-elements merely depending on the ratio of voltages. This direct relation is achieved by making use of a feedback configuration, in which the inputs are compensated using an identical, but revesely connected multiplier.

Contents

1 Introduction 1

2 Cellular Neural Networks (CNN) 3
- 2.1 Definition . . . 3
- 2.2 Structure . . . 3
- 2.3 System Operation . . . 4
- 2.4 Linear Cloning Template CNN (LCT-CNN) . . . 5
  - 2.4.1 Implementation . . . 6
  - 2.4.2 Examples . . . 7
- 2.5 2-Quadrant LCT-CNN . . . 8
- 2.6 Full Range CNN (FR-CNN) . . . 10
- 2.7 Conclusions . . . 11

3 System Architecture 13
- 3.1 Initialization of cells . . . 13
- 3.2 Edge cells . . . 14
- 3.3 Multiplier . . . 14
  - 3.3.1 Two Transistor Multiplier . . . 15
  - 3.3.2 Differential Stage Multiplier . . . 17
  - 3.3.3 Comparison . . . 18
- 3.4 Representation of Variables . . . 18
- 3.5 Interconnection . . . 19
- 3.6 Accuracy . . . 20
- 3.7 Technology . . . 21
  - 3.7.1 Process . . . 21
  - 3.7.2 Simulation . . . 22
- 3.8 Design Objectives . . . 22

4 Multiplier Design 25
- 4.1 Two transistor Multiplier Design . . . 25
- 4.2 Two Transistor Multiplier Biassing . . . 27
  - 4.2.1 Large Signal Stability . . . 28
  - 4.2.2 Small Signal Stability . . . 29
  - 4.2.3 Design . . . 31
  - 4.2.4 Simulations . . . 32
- 4.3 Conclusions . . . 34

5 Cell Core Design 35
- 5.1 State Implementation . . . 35
- 5.2 State Feedback Implementation . . . 36
- 5.3 Op-Amp Design . . . 38
  - 5.3.1 Requirements . . . 38
  - 5.3.2 Design . . . 40
  - 5.3.3 Clipping . . . 45
  - 5.3.4 Conclusions . . . 47
- 5.4 Resistor Design . . . 47

5.4.1 Requirements 48
5.4.2 Design 48
5.4.3 Sizing 50
5.4.4 Conclusions 50
5.5 Simulations 51
5.6 Conclusions 52

6 Cell and Network Control **55**
6.1 Operation Procedure 55
6.2 Feedback control 56
6.3 Input Storage 56
6.4 Cell Addressing 58
6.5 Conclusions 58

7 Total Cell Circuit **59**
7.1 Description 59
7.2 Template Transformation 59
7.3 Example: CCD 60

8 Conclusions and Recommendations **63**

Acknowledgements **65**

References **67**

Appendices:

A Transformation of the saturation function **69**

B Cmos Level-2 Parameters **71**

C Cell Core Dynamics **73**

D Total Cell Circuit **75**

E Hspice Listing CCD-test **77**

Chapter 1

Introduction

For many years now, Artificial Neural Networks (ANN) have been the subject of study in search of solutions to problems, which seem hard to tackle with the more 'common' methods, e.g. the von Neumann computer. These problems, such as pattern recognition and image processing, require a high degree of information processing and robustness, both of which can be provided by using the parallel structures of neural networks.

An artificial neural network consists of a collection of cells which are densely interconnected in a certain topology, often reminiscent of biological neural nets, e.g. the multi-layer perceptron. These so-called cells are simple computational elements, the outputs of which are determined by a, in general non-linear, weighting function of the inputs. Usually this function is a kind of saturation of a linear sum of the weighted inputs.

In 1988 Chua and Yang [1] proposed a new class of artificial neural networks called Cellular Neural Networks (CNN). The main properties of this class of networks are that the cells are locally connected (originally on a 2-dimensional grid), contain some sort of degrading memory (originally a lossy RC-integrator) and have continuously valued signals.

Despite the local connectivity, these circuits are able to perform global tasks as a result of propagation-effects, and prove to be quite effective in image-processing and other tasks which rely on these local inter-actions. This local connectivity however enables high speed and complexity operation, as the network is much easier to implement in VLSI.

Chapter 2

Cellular Neural Networks (CNN)

2.1 Definition

Since 1988, the original idea of Chua and Yang, based on the local connectivity of *cellular automata* and the cell properties of neural networks, has led to a much wider range of cellular circuits than initially proposed. In general any circuit which satisfies the following definition [3] can be called a *Cellular Neural Network* or *Cellular Non-linear Network* (CNN).

Definition: The CNN is a

- 2-, 3-, or n-dimensional array of
- mainly identical **dynamical systems**, called cells, which satisfies two properties:
- **most interactions are local** within a finite radius r, and
- **all state variables are continuously valued** signals

These local interactions are specified by a so-called (cloning) *template-set*. This template-set can be identical for each cell in the network, but does not necessarily need to be space-invariant. This template-set defines, together with the dynamics of the cell, it's behaviour in terms of the input-, output-, and state-variables of the *neighbourhood* of a cell. This neighbourhood N_r is the collection of all cells which lie within a radius of r cells from the specified cell $C(i)$, and thus includes the specified cell itself:

$$N_r(i) = \{\, C(j) \mid \mathrm{d}(i,j) \leq r \,\}$$

In this definition of a CNN *no* restrictions are made on the time-variable (whether it should be continuous or discrete), the grid-topology or the mode of operation, nor does this definition exclude any form of interaction or cell dynamics.

2.2 Structure

Although the definition of a CNN does not restrict the grid-topology or grid-dimensions (as said in the previous section), the most common net-structure is a 2-dimensional array on a square or hexagonal grid with a space-invariant template-set. The neighbourhood-size or radius seldom exceeds two and is usually one (resulting in 8 or 6 neighbours respectively). This structure (as shown in figure 2.1) is a result of the main application-fields of the CNN: pattern-recognition and image-processing, and has several advantages compared to other neural nets:

- Due to the local connectivity with a small radius, the implementation in VLSI is much easier.
- Due to the space-invariancy, the template-set needs to be stored only once per net, instead of individual storage in each cell.
- Due to the local connectivity, the net can be easily expanded by connecting the edge cells[1] of two separate nets.

[1] An edge cell is a cell which neighbourhood is not fully present in the net. For example, if the radius is two, not only the cells at the edge of the net, but also the cells with a one-cell distance to the edge are called edge cells.

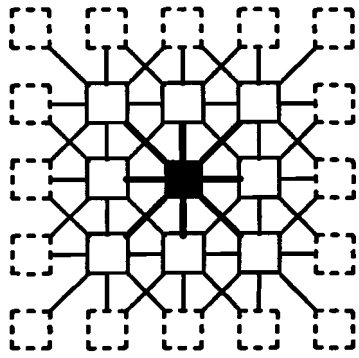

Figure 2.1: A typical net-structure with a neighbourhood-size of one.

2.3 System Operation

Since the CNN operation mainly depends on information flow in the network, rather than on the operation inside a cell, the operation of such a cell is relatively simple and can be stated in a single dynamical equation. A general form of such an equation in continuous time is given in (2.1) (see [4]).

$$\Delta_t x_i(t) = g[x_i(t)] + \sum_{k \in N_r(i)} \hat{A}_{i,k}[y_k(t);\tau] + \sum_{k \in N_r(i)} \hat{B}_{i,k}[u_k(t);\tau] + I_i(t) \qquad (2.1)$$

$$\text{with:} \qquad y_i(t) = \hat{\mathcal{F}}_i[x_i(t);\tau] \qquad (2.2)$$

In equations (2.1) and (2.2) x, y, u, and I denote respectively the cell state, output, input, and bias; g is a local feedback function and $\hat{A}$, $\hat{B}$, and $\hat{\mathcal{F}}$ denote respectively the feedback functional, the input functional, and the output functional. Such a functional $\hat{A}[y_k(t);\tau]$ is (in general) a non-linear function of y_k depending on all it's values between time $t-\tau$ and t. Hence the cell is said to have a memory with a duration τ. Finally, Δ_t denotes a differential operator. A functional diagram of the operation within a cell is shown in figure 2.2.

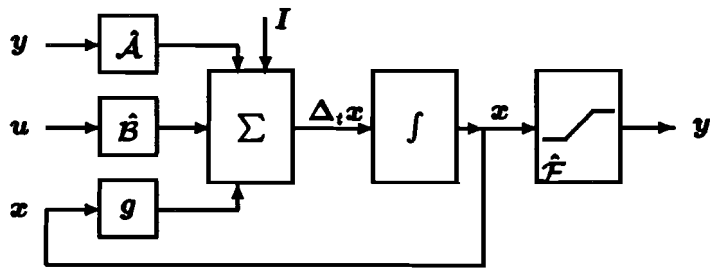

Figure 2.2: A functional diagram of a CNN cell.

Although the input and output of the cell might be obvious, it should be remembered that there are *three* ways to feed data to the network. Apart from the usual input u, the initial state $x(t_0)$ and the bias $I(t)$ can be used to supply the data. In many cases not the input u, but the initial state $x(t_0)$ is used. In that case no influence is exercized on the network *during* relaxation[2].

[2]Relaxation: This is the time during which the network relaxes to a stable output state.

2.4 Linear Cloning Template CNN (LCT-CNN)

The original CNN proposed by Chua and Yang was a CNN on a square, 2-dimensional grid with linear templates. In this case the functionals $\hat{A}$ and $\hat{B}$ are simple weight-multiplications and is $g[x(t)] = -x(t)$ a direct negative feedback. The functional $\hat{\mathcal{F}}$ consists of a piece-wise linear saturation function. In the special (but most common) case the templates are space-invariant, equations (2.1) and (2.2) can be written as

$$\frac{dx_i(t)}{dt} = -x_i(t) + \sum_{k \in N_r(i)} A_{i,k} y_k(t) + \sum_{k \in N_r(i)} B_{i,k} u_k(t) + I_i(t) \tag{2.3}$$

$$\text{with:} \qquad y_i(t) = f(x_i(t)) = \frac{1}{2}(|\, x_i(t)+1 \,| - |\, x_i(t)-1 \,|) \tag{2.4}$$

The saturation function from equation (2.4) is shown in figure 2.3.

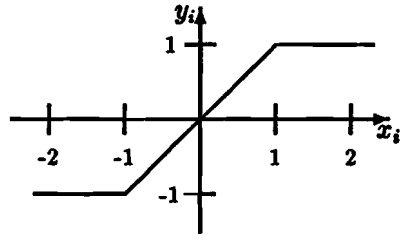

Figure 2.3: The Chua-Yang CNN output saturation function.

Under certain constraints, the stability of this type of CNN can be proved using a Lyapounov energy-function (see [1]). Since the output is restricted, these constraints apply to the input and the initial state of the cell. Furthermore, sufficient self-feedback (2.7) and a symmetric template (2.8) are required. However, the latter requirement is sufficient but not necessary, and stability can be proved for certain templates not satisfying this constraint. Other classes of templates, for which stability can be proved, include opposite-sign, positive-cell-linking, and acyclic templates (see [4]). The stability constraints for all i and k are

$$|\, x_i(0) \,| \leq 1 \tag{2.5}$$

$$|\, u_i(t) \,| \leq 1 \tag{2.6}$$

$$A_0 > 1 \tag{2.7}$$

$$A_{k,i} = A_{i,k} \tag{2.8}$$

Apart from stability the constraints (2.5) to (2.7) imply two other important properties. Due to a feedback larger than unity in each cell (2.7), the output of a cell cannot be stable unless this feedback is effectively broken by the saturation of its output. Therefore, the output will only be stable at the saturation-values (see [5]):

$$y_i(\infty) \in [+1, -1] \tag{2.9}$$

Because the output, the inputs and the initial state of a cell are limited, the range over which its state can vary is bounded as well. The maximum value of the state can be found by solving the dynamic equation (2.3) as shown in [5].

$$|x_i(t)| \leq 1 + \sum_{k \in N_r(i)} |A_{i,k}| + \sum_{k \in N_r(i)} |B_{i,k}| + |I_i| \tag{2.10}$$

2.4.1 Implementation

The cells of the network proposed by Chua and Yang were implemented using variable-gain voltage-controlled current-sources (VCCS). As a result, the summing could be easily done by leading these currents into a single node. The complete circuit of a single cell is shown in figure 2.4. The gains of the different VCCS are:

$$I_{xu,i,k} = B_{i,k}v_{u,k} \tag{2.11}$$

$$I_{xy,i,k} = A_{i,k}v_{y,k} \tag{2.12}$$

$$I_{y,i} = \frac{1}{2R_y}(\mid v_{x,i}+1 \mid - \mid v_{x,i}-1 \mid) \tag{2.13}$$

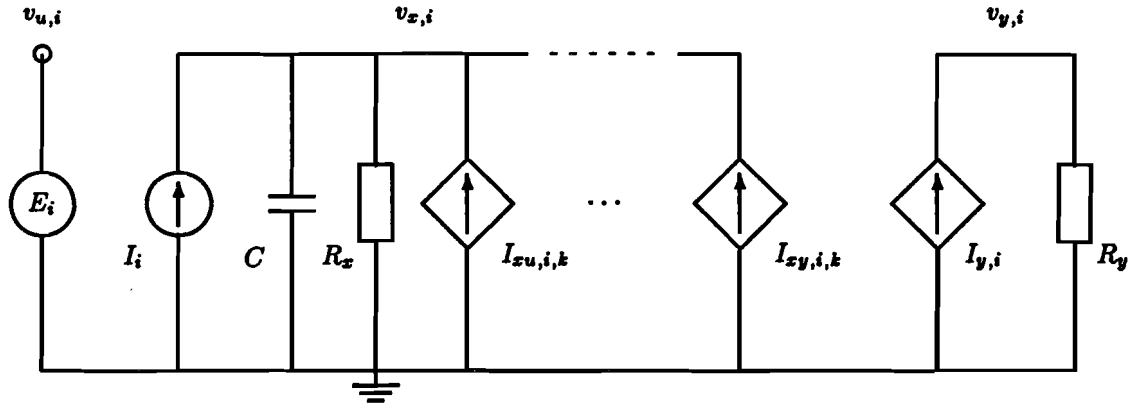

Figure 2.4: The basic cell circuit of the Chua-Yang CNN.

The dynamic state equation then becomes

$$R_xC\frac{dv_{x,i}(t)}{dt} = -v_{x,i}(t) + R_x\sum_{k\in N_r(i)}A_{i,k}v_{y,k}(t) + R_x\sum_{k\in N_r(i)}B_{i,k}v_{u,k}(t) + R_xI_i(t) \tag{2.14}$$

Comparing equation (2.14) to equation (2.3) shows that the time variable t is scaled by a factor $1/R_xC$, and that the templates have become conductances which are expressed in fractions (or multiples) of $1/R_x$.

The main problem of this implementation are the numerous VCCS (typ. 19 in a single cell), especially when the network has to have a variable template-set, and thus variable-gain VCCS. Not only the number of VCCS, but also the fact that they have to be linear in 4 quadrants often results in large circuits, and thus a large cell-area. Even though the smallest possible 4-quadrant variable-gain VCCS (or multiplier) consists of only 2 MOS-transistors in linear mode [6], the circuit requires a large biasing circuit due to the 'constant output voltage' constraint. The question is whether this 4-quadrant multiplying operation is necessary or not. If the same dynamics were to be obtained using, for example, only positive valued signals (combined with 2-quadrant templates) this would lead to multipliers which operate in just two quadrants. The circuit for a single cell would be less complex, and results in a smaller cell-area[3].

[3]Under the condition of course that no extremely large transistors are to be used.

2.4.2 Examples

To illustrate the operation of a LCT-CNN, two applications for image-processing are discussed here. The first one, hole-filling, shows that information can be supplied to the net in different ways, as mentioned in section 2.3. The second application, Connected Component Detection or CCD, shows the global information processing capabilities of a CNN. In order to easily describe the templates of a net on a square grid, the following notation using matrices is introduced. Each cell in the square grid is identified by it's coordinates x and y. The neighbourhood of the cell now consists of every cell with coordinates $x-r$ to $x+r$ and $y-r$ to $y+r$ if the neighbourhood-size is r. Each template-element belonging to a certain cell in this neighbourhood, can be identified by the relative position to the cell $C_{x,y}$. These template-elements can now be written into matrix-form. With a neighbourhood-size of $r = 1$ the feedback-template can be described by

$$A = \begin{bmatrix} A_{-1,-1} & A_{0,-1} & A_{1,-1} \\ A_{-1,0} & A_{0,0} & A_{1,0} \\ A_{-1,1} & A_{0,1} & A_{1,1} \end{bmatrix}$$

In order to visualize the input- and output-images, symbols instead of actual values will be used. As the outputs of the cell are binary, and the inputs will be too in many cases, the following method is used:

$$\text{notation of value: } v_{x,y} \quad \begin{cases} \cdot & v_{x,y} < 0 \\ \bullet & v_{x,y} \geq 0 \end{cases}$$

Holefilling is an application to "fill" enclosed regions in an image. This can be used to remove unwanted features of an object, such as printed text. These features could interfere, for example, with the recognition of the object. The image to fill is applied to the input of the cells, whilst the initial state is set to 1. The template-set used for holefilling is

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 0 \end{bmatrix}; \; B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \; I = -1$$

The input-image and the resulting output-image of the CNN are shown in figure 2.5.

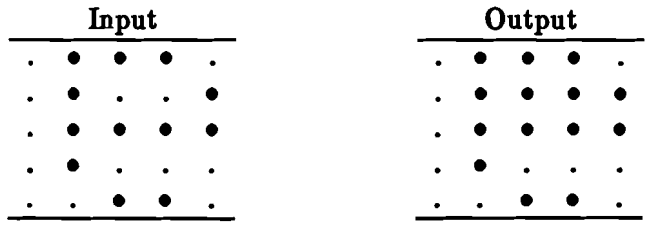

Figure 2.5: An example of holefilling.

Connected Component Detection or CCD is used as a part of an application to determine how many objects there are within the image. The template-set moves the image in a dedicated direction (in this case horizontally to the right), while shrinking every object seen on a line to a size of one. After relaxation, the output shows a pattern from which the number of objects on each line can be determined by counting the remaining dots on that line. The template-set used for CCD is

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix}; \; B = 0; \; I = 0$$

Since the input-template B only contains zero's, the only way to supply the image to the net is by using the initial state. The input-, and output-image are shown in figure 2.6

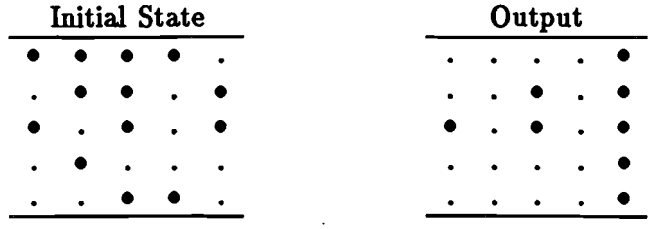

Figure 2.6: An example of CCD.

2.5 2-Quadrant LCT-CNN

An easy way to change the operation values of the signals is to change the output saturation function $f(x_i(t))$. This will not only alter the output values but also the input values (since they are exchangeable) and the state values. A general piece-wise linear saturation function, shown in figure 2.7 can be written as

$$y_i(t) = \frac{b-a}{4}\left\{\left|\frac{2}{d-c}x_i(t) - \frac{c+d}{d-c}+1\right| - \left|\frac{2}{d-c}x_i(t) - \frac{c+d}{d-c}-1\right|\right\} + \frac{a+b}{2} \qquad (2.15)$$

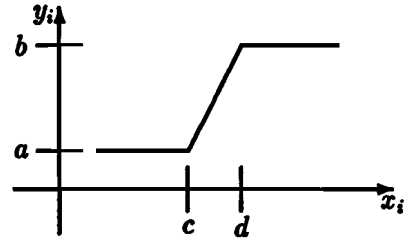

Figure 2.7: A general piece-wise linear output saturation function.

In order to evaluate the effects of this change, a transformation is used to restore the original saturation function (2.4). This transformation (see App. A) results in

$$\begin{aligned}\frac{d\left\{x_i'(t)+\frac{c+d}{d-c}\right\}}{dt} = & -x_i'(t) + \sum_{k\in N_r(i)}\left(\frac{b-a}{d-c}\right)A_{i,k}y_k'(t) + \cdots \\ & \sum_{k\in N_r(i)}\left(\frac{b-a}{d-c}\right)B_{i,k}u_k'(t) + \cdots \\ & \left\{\left(\frac{2}{d-c}\right)I_i(t) - \frac{c+d}{d-c} + \sum_{k\in N_r(i)}\left(\frac{a+b}{d-c}\right)(A_{i,k}+B_{i,k})\right\} \end{aligned} \qquad (2.16)$$

This new dynamical equation can be simplified by substituting the different template-elements:

$$\frac{d\left\{x_i'(t)+\frac{c+d}{d-c}\right\}}{dt} = -x_i'(t) + \sum_{k\in N_r(i)}A_{i,k}'y_k'(t) + \sum_{k\in N_r(i)}B_{i,k}'u_k'(t) + I_i'(t) \qquad (2.17)$$

$$\text{with} \quad \begin{cases} A'_{i,k} &= \left(\frac{b-a}{d-c}\right) A_{i,k} \\ B'_{i,k} &= \left(\frac{b-a}{d-c}\right) B_{i,k} \\ I'_i(t) &= \left(\frac{2}{d-c}\right) I_i(t) - \frac{c+d}{d-c} + \sum_{k \in N_r(i)} \left(\frac{a+b}{d-c}\right) (A_{i,k} + B_{i,k}) \end{cases} \tag{2.18}$$

Apart from the term $\frac{c+d}{d-c}$ on the left-hand side of equation (2.17), this mathematical description is identical to the original equation (2.3), defined by Chua and Yang. Although it seems that this term might be omitted, it has a strong influence on the dynamic behaviour of the system. This might become clear by again substituting $x'_i(t)$ by $x'_i(t) = x''_i(t) - \frac{c+d}{d-c}$, which shifts the term from the left-hand side to the right-hand side of the equation, but also changes the non-linearity. However, a graphical representation of the transformation shows the direct influence of the term on the cell-state $x'_i(t)$. As seen in figure 2.8, the term $\frac{c+d}{d-c}$ represents a direct offset to the cell-state, thus effectively changing the range in which the saturation function has a proportional (non-constant) transfer to the output.

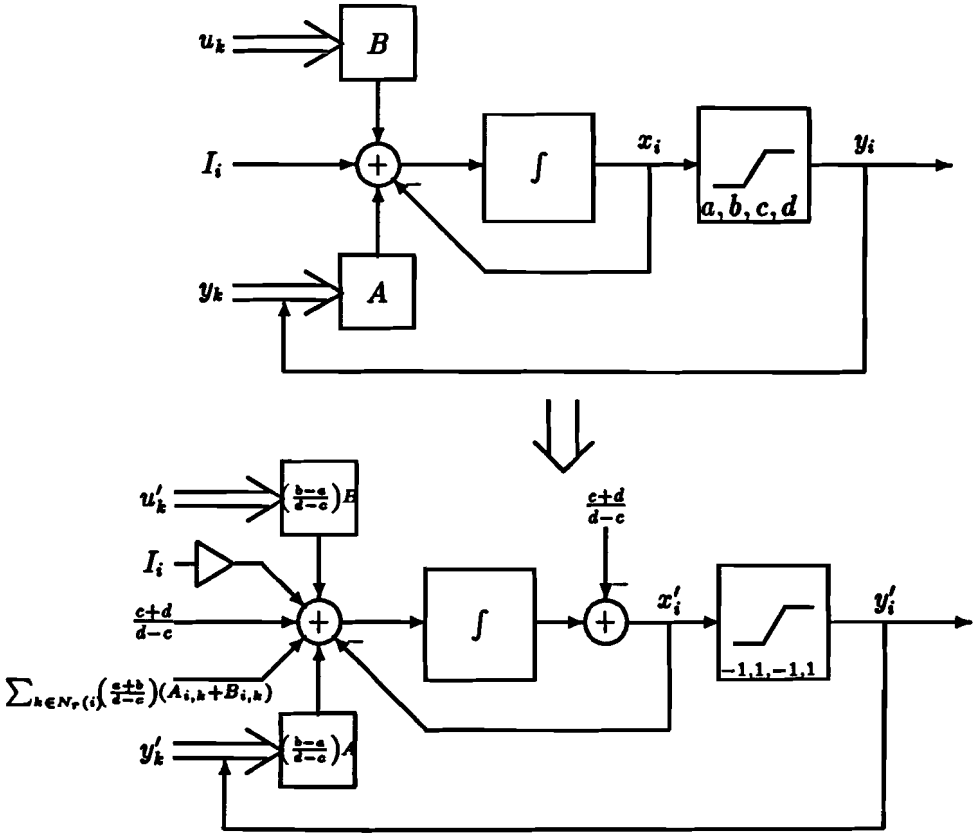

Figure 2.8: Graphical transformation of the cell to restore the old saturation function

Therefore, this term alters the dynamic behaviour of the system. Although the system is still able to function as a CNN, the templates that are to be used for a specific processing-function become unpredictable, and do not relate directly to the original templates. However, forcing this term to zero by choosing $c = -d$ does not collide with our aim of finding a CNN using only 2-quadrant multipliers, and results in an identical dynamical equation, and thus identical dynamic behaviour. Choosing a and b of the saturation function in one quadrant (both

positive or negative) results in the 2-quadrant multiplications $A'_{i,k}y'_i$ and $B'_{i,k}u'_i$. The ranges of the different signals in a cell now become

$$\begin{array}{lrcl} \text{state:} & -d & \leq |x_i| & \leq d \\ \text{input:} & a & \leq |u_i| & \leq b \\ \text{output:} & a & \leq |y_i| & \leq b \end{array}$$

Unlike the dynamic behaviour of the cell, the templates to be used for a specific operation of the CNN are changed by altering the ranges of the signals in a cell. As can be seen from (2.18), the feedback-, and input-templates A and B are scaled by a factor $\frac{b-a}{d-c}$, corresponding to the slope of the saturation-function. The bias I is not only scaled (by a factor $\frac{2}{d-c}$), but also shifted by a term proportional to the sum of the elements of the new templates A' and B'. This can be easily explained by looking at figure 2.8 again. If the inputs and outputs of a cell were to operate in a single quadrant, an offset is needed to move the upper and lower limit into the same quadrant. This offset is multiplied with every element of the templates A and B, as this offset is present in every input or output. Because this offset is constant, it effectively adds to the bias term I.

2.6 Full Range CNN (FR-CNN)

Another problem in the implementation of a CNN is caused by the large dynamic range of the cell's state. This dynamic range depends on the minimum accuracy needed for correct operation, which is at least less or equal to the input-range of the saturation-function, and the maximum state value possible. As can be seen from (2.10), the maximum state value depends on the number of neighbours (or equivalently the neighbourhood-size), and the element-values of the templates. In case of a variable-template or programmable CNN, the maximum element values should be used to determine the maximum state value, which leads to an even larger dynamic range. For correct operation of the LCT-CNN, it is necessary that the cell's state should not be limited by the physical implementation of the net (such as supply voltage or maximum current). This gives rise to the need for more accurate circuits (down-scaling the state ranges) or higher power consumption (up-scaling the physical boundaries).

To avoid these complications in implementing a CNN, Rodriquez-Vazquez et al. [2] proposed a new model for CNNs, called Full Range Cellular Neural Networks (FR-CNN). In this new model, the state of the cell is bounded by the input range of the original saturation-function, also limiting the output of the cell, and making the non-linearity or saturation-function superfluous. This new model can be described by

$$\frac{dx_i(t)}{dt} = -g[x_i(t)] + \sum_{k\in N_r(i)} A_{i,k}y_k(t) + \sum_{k\in N_r(i)} B_{i,k}u_k(t) + I_i(t) \qquad (2.19)$$

$$\text{with:} \quad g[x_i] = \lim_{m\to\infty} \begin{cases} -m(x_i+1)-1 & x_i < -1 \\ -x_i & \text{otherwise} \\ -m(x_i-1)+1 & x_i > 1 \end{cases} \qquad (2.20)$$

This new feedback function $g[x_i]$ insures that the state will no longer change ($\frac{dx_i}{dt} = 0$) if it has reached the limits of it's operation range $[-1, 1]$.

Although the system dynamics of a cell are quantitatively changed, the qualitative behaviour of the total net is not altered, as it depends on the (unchanged) interactions *between* cells. The main property of the LCT-CNN cell, namely its ability to yield (for a feedback greater than unity) two stable equilibrium points separated by an instability region, is maintained in this new

model. Moreover, other important properties such as stability and binary output values, are maintained, and there exists a one-to-one correspondence between the solutions of both models (see [7]). Furthermore, the derivation for two quadrant operation, as set forth in the previous section, remains fully intact. As the Full Range model only substitutes the direct feedback with a piece-wise-linear (pwl) function, the transformation to restore the old saturation-function will also alter this pwl-function to suit the new state-ranges.

2.7 Conclusions

As was shown in section 2.5, it is possible to use another saturation function in the CNN of Chua and Yang, without altering the dynamics. The only restriction to this function is, that the input-signal range lies symmetrically around zero (identical to choosing $c = -d$). By choosing the output-signal range to lie in one quadrant (always positive or always negative), the multiplications $A'_{i,k}y'_i$ and $B'_{i,k}u'_i$ will operate in just two quadrants. The implementation of 2-quadrant multipliers is much easier, and will result in smaller circuits (and cells) than would have been the case using 4-quadrant multipliers.

Furthermore it was shown in the previous section that, by using a Full Range model instead of the original LCT-CNN, the implementation could be made easier by reducing the dynamic range needed for correct operation. Although this new model limits the state of the cell, it does not qualitatively change the operation of the CNN. As the 2-quadrant operation of a CNN is merely a translation of operating-ranges, the Full Range model can also be applied to this type of CNN, allowing both 2-quadrant operation and improved accuracy.

Chapter 3

System Architecture

As it comes to making a VLSI implementation of a Cellular Neural Network, and in particular a programmable / variable-template CNN, more aspects have to be considered than the realization of the basic cell dynamics. Although the multiplier (or synapse or weight) will determine most of the architecture of a cell (as it is the most important and frequent part of a CNN), aspects such as intercell-connection, accuracy, and technology can play an important role in the choice of architecture and implementation. Other aspects, often forgotten when implementing a CNN, are the initialization of the cells and the influence of edge cells on the operation of the CNN.

3.1 Initialization of cells

As was mentioned in a previous chapter, the initial state-value $x_i(t_0)$ of the cells can be an important way to feed data to the network. In many applications, e.g. CCD, influence of the input *during* relaxation is not desired ($B = 0$), and the initial state is the only way to supply the input.

In fixed-application or fixed-template CNNs this initialization can be difficult to realize, if, for example, the state can't be directly identified with a single node-voltage or current, or can't be influenced directly (i.e. the state is the output of an op-amp). Furthermore, several switches are needed to break the feedback loop (to stop the network from starting the relaxation process) and to connect the state(-capacitor) to an input. However, when implementing a variable-template CNN, the question rises whether a certain template could be used to set the state. In this way, initialization could be made much easier and some switches might not be necessary. Looking at the dynamic cell equation (2.3), it is easy to see, that choosing $B_{i,i} = 1$ and setting all other template elements zero

$$\begin{aligned} A_{i,k} &= 0 && \forall_{i,k} \\ B_{i,k} &= \begin{cases} 0 & \forall_{i,k \mid i \neq k} \\ 1 & \forall_{i,k \mid i = k} \end{cases} \\ I_i &= 0 && \forall_i \end{aligned} \tag{3.1}$$

results in a straightforward first-order linear differential equation

$$\frac{dx_i}{dt} = -x_i + 1 \cdot u_i$$

The solution of this equation is an exponential rise or decay of the state value x_i from it's starting value at t_0 to the input value u_i

$$x_i(t) = u_i + [x_i(t_0) - u_i]e^{-(t-t_0)}$$

Although the state theoretically never will reach the input value, it will be within a certain accuracy of this value u_i after a few seconds[1].

Realizing the initialization of the net in this manner brings along a few drawbacks. Since the feedback loop is not broken during initialization, the template-set and inputs have to be switched to their new (operating) value simultaneously. This gives rise to some problems in the

[1]More general: after a few times the time-constant of the cell. In a physical implementation of a CNN the time-variable is scaled with a certain time-constant.

control of the CNN. Furthermore, as all cells are initialized simultaneously, it is necessary that the inputs of all cells are available. In case of small CNNs (typ. up to 40 cells) this isn't a problem. Larger CNNs often use a kind of cell-addressing to write the inputs and to read the outputs (using on-chip storage of inputs; the outputs are 'stored' by the feedback) in order to reduce the number of pins on a chip. As long as the inputs stored on-chip contain the initial state value, the states are maintained. However, the inputs have to be set-up for their operating-values u_i. Since not all inputs are changed simultaneously, the initial states of some cells may start to change, regardless which template-set is used.

If a switch is used to restrain the network from starting the relaxation (effectively breaking the feedback loop), these problems are solved, allowing this method of initializing to be used for networks in which the state cannot be accessed or identified directly.

Finally, it should be said that, although proved for an LCT-CNN, this method is applicable for any type of variable-template CNN, including the 2-Quadrant LCT-CNN.

3.2 Edge cells

Edge cells are cells which neighbourhoods are not fully present in the net, or, put in another way, don't have a fully occupied neighbourhood. Therefore, these cells actually have another template-set: template-elements belonging to non-existing neighbourhood cells appear to be zero. However, the same result would be obtained if these neighbourhood cells were present having an input and output equal to zero: $u_i = 0$ and $y_i = 0$. Implementing a template-element (or multiplier) in a LCT-CNN and connecting it to zero or not implementing it at all would result in exactly the same operation of the edge cell.

Not implementing the template-elements in a 2-Quadrant LCT-CNN however, would lead to incorrect biasing and thus incorrect operation of the edge cell. As the input and output ranges of the cell are transformed, the original bias-points of the 'ghost' neighbourhood cells, $u_i = 0$ and $y_i = 0$, are also transformed to the new bias-points $u_i' = y_i' = \frac{a+b}{2}$. Therefore, the template-elements (or multipliers) belonging to non-existing neighbourhood cells should be implemented having a constant input equal to $\frac{a+b}{2}$. These constant inputs could be realized by using the outputs of dummy cells, or dummy edge cells, which have no connections to any other cell (equivalently: have a template-set $A = 0$ and $B = 0$).

This translation should also be taken into account in case these edge-cells *are* connected to, for example, special bias-voltages or dummy-cells. For the Connected Component Detection template, the edge-cells are usually connected to a special bias-voltage representing the non-existing neighbourhood cells having a 'low' output.

3.3 Multiplier

To transform the CNN model equations (2.3) or (2.17) into an equivalent circuit each variable has to be represented by a voltage or a current. As current summing is easy using Kirchhoff's current law (feeding all currents into a single node with one output connection), the variable I_i and the results of the multiplications $A_{i,k} \cdot y_i$ and $B_{i,k} \cdot u_i$ can best be represented by currents to avoid unnecessary conversions from voltages to currents. Since these multipliers are numerous in each cell (two for each neighbour), it is important that these multipliers are as small as possible to allow small cell-design and high cell-density. For 2-quadrant operation in a CMOS implementation two multipliers having current output are considered. Finally, the input and output of a cell should preferably be represented by a voltage as these signals have to be dispersed to each cell in the neighbourhood.

3.3.1 Two Transistor Multiplier

The two transistor multiplier (or 2T-multiplier for short) consists of two MOS transistors in linear mode with a common source node, as depicted in figure 3.1.

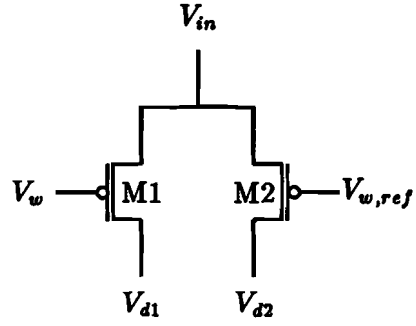

Figure 3.1: A Two Transistor Multiplier in PMOS.

The drain-currents of the transistors are

$$I_{d1} = \frac{\beta_1}{2}\left[2(V_{in} - V_w + V_{T1})(V_{in} - V_{d1}) - (V_{in} - V_{d1})^2\right](1 + \lambda(V_{in} - V_{d1})) \tag{3.2}$$

$$I_{d2} = \frac{\beta_2}{2}\left[2(V_{in} - V_{w,ref} + V_{T2})(V_{in} - V_{d2}) - (V_{in} - V_{d2})^2\right](1 + \lambda(V_{in} - V_{d2})) \tag{3.3}$$

Using the substitutions

$$\begin{array}{rclcrcl} \beta_1 & = & \beta & & \beta_2 & = & \beta(1+\delta_\beta) \\ V_{T1} & = & V_T & & V_{T2} & = & V_T(1+\delta_T) \\ V_{d1} & = & V_d & & V_{d2} & = & V_d(1+\delta_d) \end{array} \tag{3.4}$$

and neglecting higher-order errors the *difference* between the two drain-currents can be approximated by

$$\begin{aligned} \Delta I = I_{d2} - I_{d1} \approx\ & \beta(V_w - V_{w,ref})(V_{in} - V_d)(1 + \lambda(V_{in} - V_d)) \cdots \\ & + \delta_\beta \cdot \frac{\beta}{2}(V_{in} - 2V_{w,ref} + 2V_T + V_d)(V_{in} - V_d) \cdots \\ & + \delta_T \cdot \beta V_T(V_{in} - V_d) \cdots \\ & - \delta_d \cdot \beta V_d(V_d - V_{w,ref} + V_T) \end{aligned} \tag{3.5}$$

In the ideal case that the transistors are identical, having the same threshold voltage V_T and transconductance β, neglecting the channel modulation λ and assuming identical drain-voltages V_d, the difference in drain-currents is proportional to the product of difference in gate-voltages and the source-drain voltage:

$$\Delta I = \beta \cdot (V_w - V_{w,ref}) \cdot (V_{in} - V_d) \tag{3.6}$$

Although theoretically this is four quadrant multiplication, as both terms can become both positive and negative, a practical implementation is hard to realize, as the currents are able to flow in two directions, depending on the drain-source voltage. In particular keeping the drain-voltages equal on a constant level is hard to implement. A two quadrant operation (with $V_{sd} \geq 0$) however is easy to implement as the currents now will flow in one direction only.

Apart from errors due to mismatching of the transistors and voltages there are some deviations / properties which have to be taken in into account:

1. The *channel length modulation* creates a non-linear error for the drain-source input $V_{in}-V_d$, as is shown in (3.5). This factor λ becomes smaller if the length of the transistors is increased. Although depending on the technology used, this error can usually be neglected if the length is more than two or three times the minimum design length of a transistor.

2. Because the bulk-source-voltage of both transistors is variable, the threshold voltage changes due to the *body-effect*. Although this doesn't change the *difference* in currents in first order, it does change the currents itself. This effect can be reduced by connecting source and bulk together. This however requires a separate well for each multiplier in a cell, as each multiplier has a different source-voltage V_{in}, causing an increase in the total layout size.

3. As the two transistors have to operate in linear mode at any time, some restrictions to the gate-voltages apply:

$$\begin{aligned} V_w &\leq V_d + V_T \\ V_{w,ref} &\leq V_d + V_T \\ \text{with: } V_T &< 0 \end{aligned}$$

4. Due to the linear operation of both transistors, resulting from high gate-source voltages, large currents will flow through the transistors, causing a large power dissipation. As always, a trade-off between power dissipation, chip area and accuracy exists. The total current flowing through the multiplier

$$I_{tot} = \beta(V_{in} + V_d + 2V_T - V_w - V_{w,ref})(V_{in} - V_d) \tag{3.7}$$

   can be decreased by

   - decreasing the transconductance by decreasing the ratio between transistor width and length $\frac{W}{L}$.
   - decreasing the operating range of $V_{in} - V_d$.
   - increasing the minimum gate-voltage $V_{w,min}$ by decreasing $|V_w - V_{w,ref}|_{max}$.
   - choosing the operating ranges closer together by decreasing $V_d - V_{w,ref}$.

   Another aspect related to the design is the ratio between the total current and the maximum current difference. As this ratio increases, the total accuracy decreases, as the output of the multiplier consists of the *difference* of two signals. Subtracting two large currents in order to find a small difference results in poor accuracy. The ratio $\frac{I_{tot}}{\Delta I_{max}}$ should be chosen as small as possible.

5. The multiplier has, as mentioned earlier, a very low output impedance and special circuitry is needed to keep the two drain voltages at a constant level. While one of the inputs is a differential voltages across two gates, having a very high input-impedance, the other is connected to the source of both transistors. This causes this input to be low impedant, and a driver circuit with low output impedance is therefore needed.

3.3.2 Differential Stage Multiplier

The differential stage multiplier consists of a simple differential pair which tail-current is controlled by a transistor in saturated mode, as shown in figure 3.2.

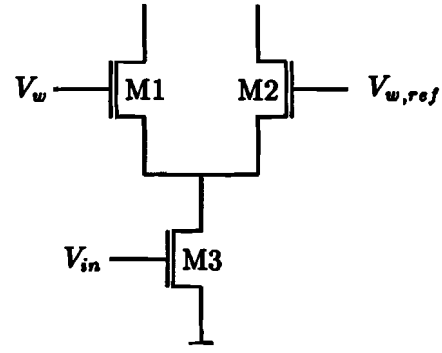

Figure 3.2: A differential stage multiplier using NMOS transistors.

Within the operating range for the gate-voltages,

$$-\sqrt{2I_{d3}/\beta_{1,2}} \leq V_w - V_{w,ref} \leq \sqrt{2I_{d3}/\beta_{1,2}}$$

the drain-currents in the identical transistors of the differential pair are, if the channel length modulation is neglected, equal to:

$$I_{d1} = \frac{1}{2}I_{d3} + \frac{1}{2}\sqrt{\beta_{1,2}}(V_w - V_{w,ref})\sqrt{I_{d3} - \frac{\beta_{1,2}}{4}(V_w - V_{w,ref})^2} \tag{3.8}$$

$$I_{d2} = \frac{1}{2}I_{d3} - \frac{1}{2}\sqrt{\beta_{1,2}}(V_w - V_{w,ref})\sqrt{I_{d3} - \frac{\beta_{1,2}}{4}(V_w - V_{w,ref})^2} \tag{3.9}$$

In which the tail-current is

$$I_{d3} = \frac{\beta_3}{2}(V_{in} - V_{T3})^2(1 + \lambda_3 V_{d3}) \tag{3.10}$$

Again using the substitutions (3.4), neglecting higher-order errors and channel length modulation effects, the difference between the two currents can be approximated by

$$\begin{aligned}\Delta I = I_{d2} - I_{d1} \approx\ & \sqrt{\frac{\beta\beta_3}{2}}(V_{in} - V_{T3})(V_w - V_{w,ref})\sqrt{1 - \frac{\beta}{2\beta_3}\frac{(V_w - V_{w,ref})^2}{(V_{in} - V_{T3})^2}} \cdots \\ & + \delta_\beta \cdot \left[I_{d3} - \sqrt{\beta}(V_w - V_{w,ref})\sqrt{I_{d3} - \frac{\beta}{4}(V_w - V_{w,ref})^2} \right] \cdots \\ & + \delta_T \cdot V_{T1,2}\sqrt{\beta}\sqrt{I_{d3} - \frac{\beta}{4}(V_w - V_{w,ref})^2}\end{aligned} \tag{3.11}$$

Even if the differential pair transistors are identical, having the same transconductance and threshold voltage and no channel length modulation, the resulting difference current suffers from non-linearities. These non-linearities can only be neglected if the gate-voltages of the differential pair are operated well within the defined operating range:

$$I_{d3} \gg \frac{\beta}{2}(V_w - V_{w,ref})^2 \tag{3.12}$$

effectively causing a high bias current in both transistors. The difference-current then becomes

$$\Delta I \approx \sqrt{\frac{\beta\beta_3}{2}}(V_{in} - V_{T3})(V_w - V_{w,ref}) \tag{3.13}$$

As the current in the tail-transistor M3 can flow in just one direction, this circuit is a true two-quadrant multiplier. Apart from the aspects concerning the body-effect and channel length modulation, mentioned with the 2T-multiplier in the preceding paragraph, the following properties should not be overlooked.

1. As mentioned, strict boundaries (3.12) for the differential gate-voltage apply, to insure linear operation. However, the input range of the gate-voltage of the tail-transistor M3, is also restricted, as the transistor has to remain in saturated mode for correct operation. The maximum gate-voltage depends on the transistor design ratios $\frac{W}{L}$, and the minimum voltage at the gates of M1 and M2:

$$V_{g3} \leq \left(\frac{\sqrt{\beta}}{\sqrt{\beta}+\sqrt{\beta_3}}\right)(V_{w,MIN} - V_{T1,2}) + V_{T3}$$

2. Unlike the 2T-multiplier, the differential stage multiplier has a high differential output impedance and high input impedance for both inputs (as all input nodes are gates).

3.3.3 Comparison

When comparing two multipliers, several aspects have to be taken into account, concerning linearity, input and output ranges, input and output impedances, chip area and power consumption. Comparing both output equations (3.6) and (3.13) makes clear that the differential current output-range for both multipliers will be approximately the same[2], as both voltage input-ranges of both multipliers will be in the order of several hundreds of milliVolts to one Volt. As both multipliers need large bias currents for correct operation, the power consumption will be comparable. However, slightly larger transistors are needed for the 2T-multiplier for equal bias currents, as they operate in linear mode. Because of this, and the presence of two current-conveyors in each cell when using the 2T-multiplier, the chip areas of both multipliers will be roughly the same. The main differences between the multipliers therefore lie in linearity and input and output impedances. For the latter, the diff. stage multiplier will be the best choice, as no input currents are needed and the output has a very high differential impedance. However, with regard to linearity, the 2T-multiplier will be a better choice, as it has less non-linear distortion for the same input- and output-ranges and equal power consumption.

3.4 Representation of Variables

As said in the previous section, each variable (input, output and state) has to be represented by a voltage or a current. Whereas the output of both multipliers are (differential) currents, the inputs of both multipliers are voltages. The two-quadrant voltage-input of a multiplier needs to be used for the template-elements, as these have not been changed into single quadrant operation. The single quadrant voltage-input will be used for either an output or an input of a cell. Input u and output y should therefore be represented by voltages. Moreover, interconnection will be much easier as voltages can be distributed more efficiently than currents, as currents have to

[2]When realized in the same polarity (NMOS or PMOS).

duplicated for each connected cell. The choice for the representation of the state is determined by the implementation of the dynamics of a cell. The integration can be done by establishing a difference in voltage between the terminals of a inductance resulting in a current, or feeding a current through a capacitor resulting in a voltage. As the signal to be integrated is already a sum of currents and capacitors are far more easy to implement in VLSI than inductances, a capacitor will be used. The resulting voltage represents the state of the cell. The dynamic equation can now be written as

$$C\frac{dv_{x,i}}{dt} = G \cdot v_g[v_{x,i}] + \sum_{k \in N_r(i)} G_{A,i,k} \cdot v_{y,k} + \sum_{k \in N_r(i)} G_{B,i,k} \cdot v_{u,k} + i_i$$

In this physical implementation, the self-feedback $v_g[v_{x,i}]$ results in a time-constant for the dynamics equal to $\tau = C/G$. This, of course, does not change the operation of the cell or the entire net, but merely changes the speed with which the net operates. The template-elements are expressed in multiples or fractions of G.

3.5 Interconnection

Because every input and output of a cell has to be distributed to it's neighbourhood (as the cell itself belongs to the neighbourhood of all cells in it's own neighbourhood), the total number of connections NOC for a single cell amounts to

$$NOC = 2 \cdot |N_r|$$

This total number of connections might be decreased by not distributing u_i and y_i separately, but combining these signals into a single signal to each neighbourhood cell, and distributing $A_{k,i} \cdot y_i + B_{k,i} \cdot u_i$ to cell k. Adding these two signals is easy as they are represented by currents, hence no extra adding circuitry is required. This change in interconnection alters the cell structure, as schematically shown in figure 3.3(b). Figure 3.3(a) shows the original configuration for a cell i having two neighbours. In these figures, the box marked with i represents the cell-core containing the integration- and non-linearity sub-circuits.

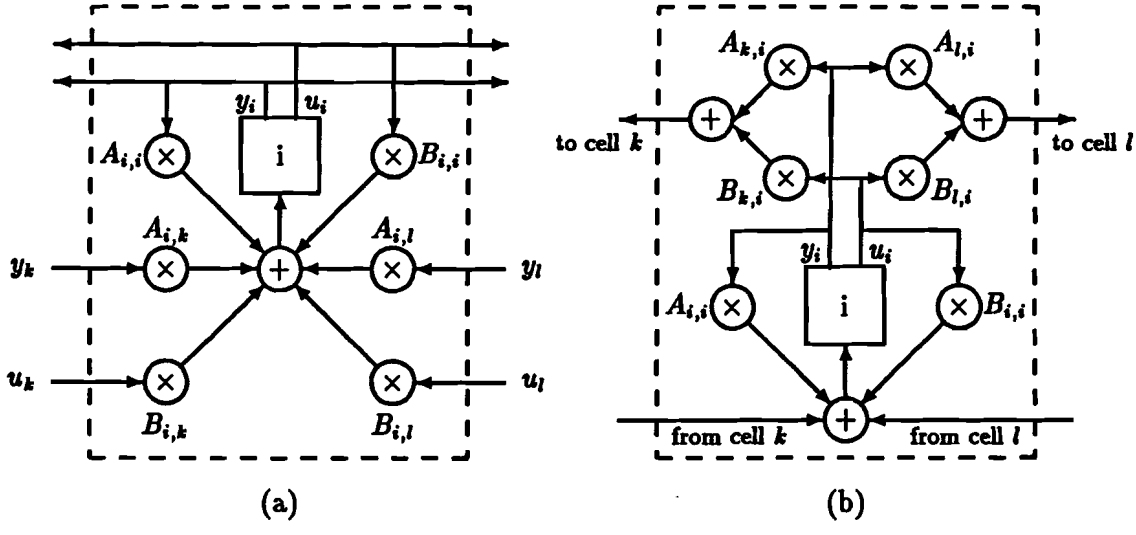

Figure 3.3: (a) Original cell structure. (b) Modified structure having less inter-cell connections.

Although not immediately obvious, this change in interconnection causes the need for extra circuitry when using the multipliers described in section 3.3. As both multipliers have differential output, no improvement in the number of connections would be made unless the output is changed to a single-ended configuration using a current-mirror. This would amount to at least two transistors extra for each pair of multipliers $A_{i,k}$ and $B_{i,k}$, depending on the current-mirror used. For the 2T-multiplier the result is more dramatic as the special circuitry for keeping the drain-voltages at constant and equal level now has to be implemented for each pair of multipliers instead of a single implementation for all multipliers $\sum_{k \in N_r(i)} A_{i,k}$ and $B_{i,k}$ together. It is therefore not recommended to use this modified interconnection structure when using the 2T-multiplier.

3.6 Accuracy

Unlike in classical linear systems the accuracy-requirements for the implementation of neural networks are hard to establish due to hard-limiting non-linearities and complex *mutual* interactions between the building blocks of a such a network. Usually, neural networks are robust against non-ideal behaviour, e.g. mismatching, as a result of a learning process, in which erratic behaviour of individual cells is compensated. For CNNs however, no learning algorithms are used most of the time and certain network behaviour is obtained by determination of the correct template-set. Even if a learning algorithm is used, less errors can be compensated, as the network has fewer variables or degrees of freedom to be controlled. In CNNs errors occur due to

- space-variant fluctuations in the template-set.
- inaccuracies in the initial state.
- inaccuracies in the output of cell, e.g. voltage drops due to output currents.
- space-variant fluctuations in the time-constant.

As the CNNs behaviour is changed by using a different template-set, the influence of non-ideal behaviour also changes. This results in different accuracy-requirements for different template-sets. In order to estimate the required accuracy for a certain template-set the following method is used (see [8]). With all it's neighbours remaining constant, the dynamic routes[3] are drawn for all possible combinations of its neighbours states. Depending on the application (or template-set), there are several *critical* dynamic routes. These are the routes for a combination of the neighbours states, which determine the operation boundaries, i.e. the boundaries between which the net should or should not change the state's value. This will be demonstrated for the Connected Component Detection (CCD) template-set as introduced in section 2.4. First, the dynamic cell-equation is, with k the sum of all external influences and the threshold I, written as

$$\frac{dx_i}{dt} = -x_i + A_{i,i} \cdot y_i + k$$

For the CCD template-set each cell has two neighbours, $A_{i,left} = 1$ and $A_{i,right} = -1$. The self-feedback $A_{i,i} = 2$, and the input-template B and threshold I are zero. For all eight possible initial-state combinations of the three cells, k can become -4,-2,0,2 and 4. The critical dynamic route is for $k = 0$, corresponding to a initial state of -1, a left neighbour of 1, and a right neighbour of -1; or the same situation, but with opposite signs. This route is shown in figure 3.4.

[3] A dynamic route is a collection of all possible combinations of dx/dt and x. Such a route, or line, can be drawn on graph with x along the x-axis and dx/dt along the y-axis. Usually, the direction in which the state moves (depending on dx/dt) is indicated by arrows.

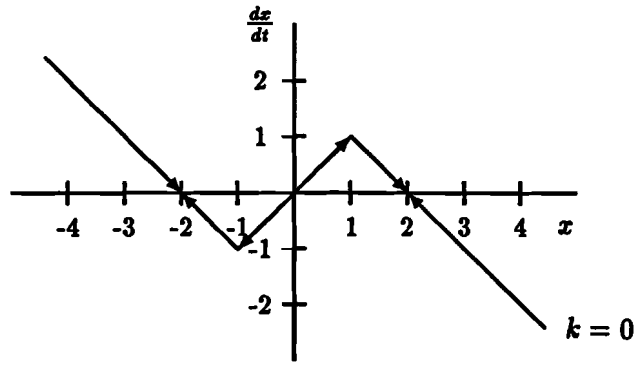

Figure 3.4: The critical dynamic route for a CCD template-set.

The two stable equilibrium points are $x = -2$ and $x = 2$. If there would be a distortion in k, resulting in vertical shift of the dynamic route, the cell will still operate correctly if two stable points are retained. As a result, variation in k should not exceed one: $-1 < k < 1$. This gives a relative accuracy of about 15% (worst case) for the contributions of the templates and outputs of the different cells.

For a correct operating network, it is necessary that each cell is working properly. This method thus gives minimum required accuracy for correct *cell* operation. However, in a true CNN all cells are active and interactions between cells have to be taken into account, as well as fluctuations in time-constants, which influence these interactions. Using computer simulations of a CNN, the minimum required accuracy can be estimated, by making space-variant fluctuations in the template-set. This shows that the estimate obtained by the preceding analytic method usually is an over-estimate, and that the true required accuracy will be somewhere between 1 and 10 percent, depending on the template-set used. The required accuracy mainly depends on the number of neighbours and the degree of self-feedback.

3.7 Technology

3.7.1 Process

The technology used for implementation of the CNN will be CMOS N-well process. As the CNN will be manufactured at IMEC in Leuven, Belgium the following limitations / properties apply:

- Double poly-silicon layer.
- Double metal layer.
- NMOS transistors have a fixed bulk-voltage; PMOS transistors are implemented in a seperate N-well and therefore have a variable bulk-voltage.
- A 2.4 μm technique will be used. The lengths and widths of the transistors will be at least 2.4 μm and can be increased using discrete steps of 0.4 μm.
- Floating-gate devices are not available.
- As ion-implant is impossible, only enhancement MOSTs can be used. This implies:

$$\begin{array}{ll} \text{NMOS:} & V_T > 0 \\ \text{PMOS:} & V_T < 0 \end{array}$$

3.7.2 Simulation

To verify the design of the CNN and to adjust the final transistor sizings, simulation software is used. For circuit-simulation a version of HSPICE (H9007) was used in combination with the graphic display tool GSI, both created by Meta-Software, Inc. As with other versions of SPICE, a circuit-description is made using node-numbers and device-models. For the description of the MOS devices, created in the IMEC process, a so-called Level-2 device parameter-set is used, based on process-parameters. In appendix B the parameters for both NMOS and PMOS transistors, capacitances and resistors and are given.

For dynamic simulations (as opposed to static analysis) layout structures need be known, as parasitic capacitances depend on transistor-size (including drain- and source-areas) and mutual position. In particular, to calculate the capacitances between the transistor and the bulk or body, drain- and source-areas (AD and PD respectively) and drain and source-perimeters (PD and PS resp.) need to be given. Using a geometry of a minimum sized transistor as shown in figure 3.5 these parameters can now be calculated according to:

$$\left.\begin{array}{ll} AS = AD = (5.6)^2 + W \cdot 1.6 & \mu m^2 \\ PS = PD = 25.6 & \mu m \end{array}\right\} \; W \leq 5.6\mu m$$

$$\left.\begin{array}{ll} AS = AD = W \cdot 7.2 & \mu m^2 \\ PS = PD = 2 \cdot (W + 7.2) & \mu m \end{array}\right\} \; W > 5.6\mu m$$

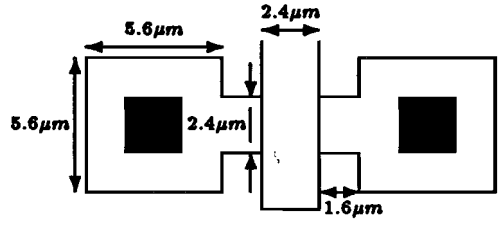

Figure 3.5: Geometry of a minimum sized MOST.

3.8 Design Objectives

As in any electronic implementation there are, apart from realizing the desired functionality, several design objectives or requirements, e.g. minium power dissipation. And there are of course the usual trade-offs between several requirements. The main trade-off is between power dissipation, accuracy, response time and chip area. More power dissipation can improve accuracy and response time, but usually increases chip area, as wider transistors need to be used. More accuracy (in terms of input- and output-ranges) can be obtained by using longer transistors, thus increasing chip area. The optimal trade-off between these properties is not fixed, and changes with the main requirements of the design.

Main Objective: Design of a Cellular Neural Network,

1. with a fully, analogue programmable template-set.
2. on a square grid.
3. with a neighbourhood-size of one, resulting in eight neighbours.

And the secondary objectives and requirements concerning the circuit-design are

Secondary Objectives:

- Maximum input- and output-ranges.
- Minimum power dissipation.
- Maximum speed or minimum response time.
- Minimum chip area.
- Minimum susceptibility to distortions, e.g. noise and parameter variations.

Chapter 4

Multiplier Design

In this chapter a design will be presented for the two-transistor multiplier or 2T-multiplier, including the necessary bias-circuits. The choice for this multiplier instead of the differential-stage multiplier, is based on the inherent linear properties of the 2T-multiplier, compared to the *linearized* properties of latter multiplier. However, as both multipliers have the same inputs (differential voltages) and output (differential current), the cell-design, described in the next chapter, is independent of the choice between these two multipliers, and both multipliers could be used.

4.1 Two transistor Multiplier Design

Before the transistors of the 2T-multiplier can be sized, several choices have to be made. The most important choice is whether the multiplier will be used in two- or four-quadrant operation. As was mentioned in the previous chapter, biasing becomes very difficult when using the multiplier in four quadrants. Therefore, a design will be made for two-quadrant operation. The different inputs and outputs of the multiplier are shown in figure 4.1.

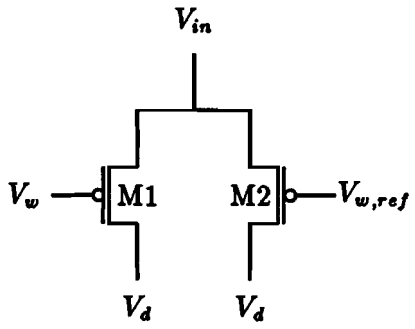

Figure 4.1: Two transistor multiplier.

Technology: First the technology in which the multiplier will be implemented is determined. The transistors of the multiplier operate in linear mode, causing a low impedance between drain and source. This results in considerable currents, which can only be decreased by lowering the trans-conductance β of the transistors. Assuming the same transistor-sizing this trans-conductance is lower for PMOS transistors. The multiplier will therefore be implemented in PMOS.

Output Range: When determining the output range, the main criterion is the maximum power dissipation or the maximum current of a single multiplier. Using a square grid, each cell has at least 19 multipliers. The maximum power dissipation in the multiplier-section of a cell then equals

$$P_{tot,mul} = 19 \cdot V_{dd} \cdot I_{max}$$

Using a power supply of 5V, and a maximum multiplier current of 10 μA, the maximum power dissipation in the multiplier-section is less than 1mW. The average power dissipation will of course be less, depending on the average multiplier-input V_{in}.

Input Ranges: When determining the input-ranges, an important aspect is the ratio between the total maximum current and the maximum current-difference $I_{max}/\Delta I_{max}$, as was mentioned in the previous chapter. To minimize this ratio, the weight-input voltage-range $V_w - V_{w,ref}$ should be as large as possible, and the drain bias-voltage V_d should be as close as possible to $V_{w,ref}$ while retaining linear operation. Choosing $|V_w - V_{w,ref}| \leq 1V$, with $V_{w,ref} = 1V$ and $V_d = 3.5V$ results in a ratio of approximately $I_{max}/\Delta I_{max} \approx 5$. The maximum input-voltage $V_{in,max}$ should be chosen as large as possible, to decrease the susceptibility to distortions. However, this input is connected to an input or the *clipping* output of a cell. To be able to create a clipped output-value independent of the output current, a small margin to the power supply voltage should be used. The maximum input-voltage is therefore set to $V_{in,max} = 4.5V$.

Summary: For the 2T-multiplier design, the following applies:

- Implementation in PMOS.
- Maximum Multiplier current $I_{max} = 10\mu A$.
- Weight Input range $0V \leq V_w \leq 2V$.
- Weight reference input $V_{w,ref} = 1V$.
- Drain bias voltage $V_d = 3.5V$.
- Input Range $3.5V \leq V_{in} \leq 4.5V$.
- Sizing of the transistors: $\frac{W}{L} = \frac{4.8\mu m}{27.2\mu m}$.

Simulations: The multiplier characteristics are shown in figure 4.2 and figure 4.3.

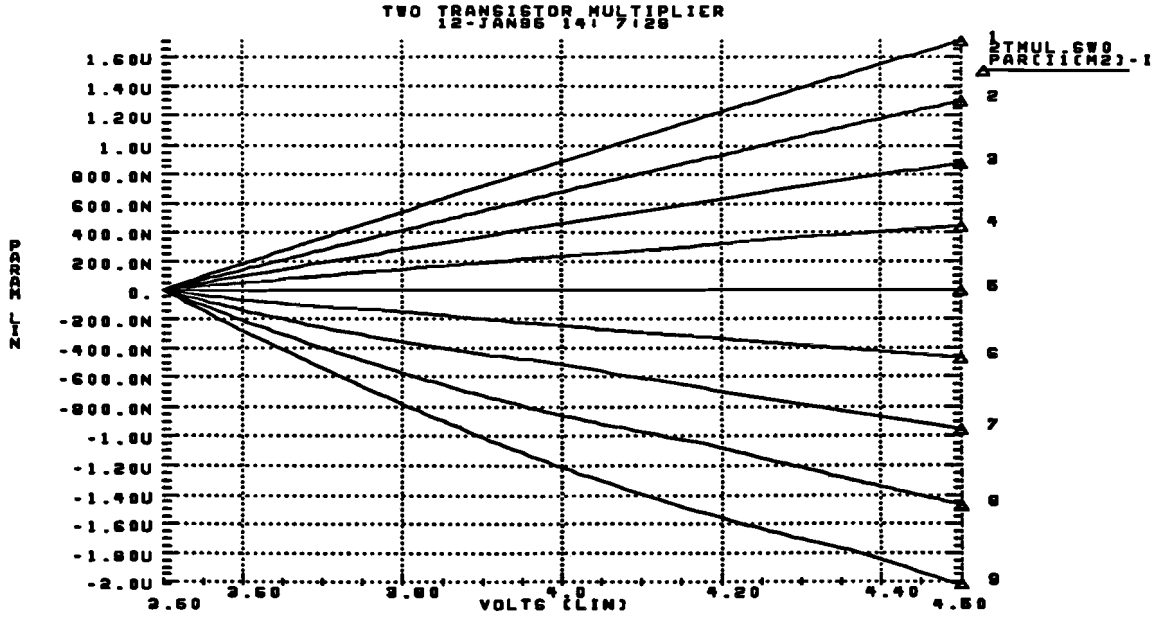

Figure 4.2: The difference-current of the 2T-multiplier.

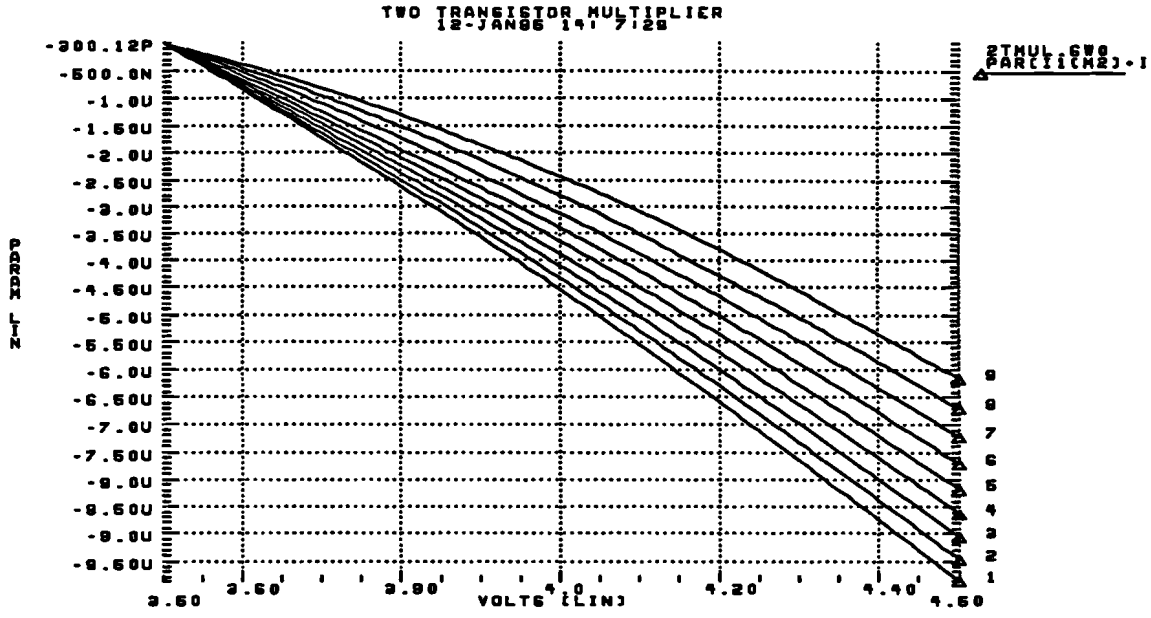

Figure 4.3: The total current of the 2T-multiplier.

4.2 Two Transistor Multiplier Biassing

In order to be able to use the 2T-multiplier, introduced in the previous section, some circuitry has to be designed for correct biasing. The two current output nodes need to be biased with a constant and equal voltage to insure linear operation. To do so, a so-called current-conveyor can be used to drain the two output currents. In general, a current-conveyor is a circuit having a current input- and output node and an input-node to control the input voltage, as shown in figure 4.4.

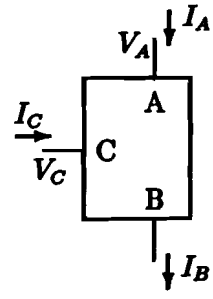

Figure 4.4: A general model of a current conveyor.

One of the input terminals, terminal C, is used to control the voltage at the other input terminal A. This controlling input can both be current or voltage. The current at the output B now solely depends on the input-current at terminal A, and is usually identical to this input-current. A straight-forward approach to this problem is using an op-amp in unity-feedback configuration, together with a transistor to control the input impedance at terminal A. This is shown in figure 4.5(a). If transistor M2 is operating in saturation, the drain-current will nearly be independent of the drain-voltage.

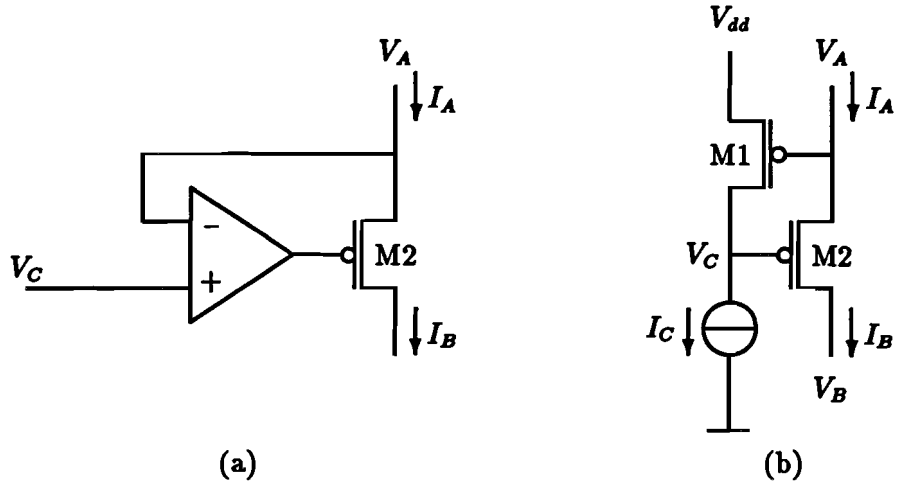

Figure 4.5: Two examples of current-conveyors.

Although the op-amp might consist of just a single differential stage, the conveyor still comprises at least six transistors in total. As this circuit has to be implemented at least twice in each cell, it leads to a considerable overhead on the actual implementation of the multipliers. However, replacing this op-amp with a single inverter, as shown in figure 4.5(b) results in the same configuration having negative feedback. In the absence of a voltage-input terminal to determine the voltage at input A, the current I_C can be used to control the source-gate voltage of transistor M1 and thus the input-voltage at terminal A, assuming that this transistor is in saturation. The gate-drain junction of transistor M1 is connected in parallel with the source-gate junction of transistor M2, ensuring that the drain-voltage of M1 will be lower than it's gate-voltage and that transistor M1 always will operate in saturation.

4.2.1 Large Signal Stability

One of the most important properties of the current-conveyor is, of course, the voltage-stability at the current input-terminal A. In particular, the influence of I_A, V_B and I_C are of interest. Therefore, the channel-length modulation will be taken into account in the evaluation of the input-voltage V_A. Using the drain-current equations for the two transistors, voltages V_A and V_C can be written as

$$V_A = V_{dd} + V_T - \sqrt{\frac{1}{\beta_1}}\sqrt{\frac{2I_C}{1+\lambda_1[V_{dd}-V_C]}} \tag{4.1}$$

$$V_C = V_A + V_T - \sqrt{\frac{1}{\beta_2}}\sqrt{\frac{2I_A}{1+\lambda_2[V_A-V_B]}} \tag{4.2}$$

Assuming small channel-length modulation $\lambda \ll 1$, the roots in the previous equations can be approximated by $\sqrt{\frac{1}{1+x}} \approx \sqrt{1-x} \approx 1-\frac{1}{2}x$, resulting in

$$V_A \approx V_{dd} + V_T - \sqrt{\frac{2I_C}{\beta_1}}\left(1-\frac{1}{2}\lambda_1[V_{dd}-V_C]\right) \tag{4.3}$$

$$V_C \approx V_A + V_T - \sqrt{\frac{2I_A}{\beta_2}}\left(1-\frac{1}{2}\lambda_2[V_A-V_B]\right) \tag{4.4}$$

Now substituting (4.4) into (4.3) and neglecting higher-order effects[1] the input-voltage V_A can be calculated:

$$V_A \approx V_{dd} - \left[1 - \frac{\lambda_1}{2}\sqrt{\frac{2I_C}{\beta_1}}\right]\sqrt{\frac{2I_C}{\beta_1}} + \left[1 - \lambda_1\sqrt{\frac{2I_C}{\beta_1}}\right]V_T + \lambda_1\sqrt{\frac{I_C I_A}{\beta_1\beta_2}}\left[1 + \frac{\lambda_2}{2}V_B\right] \quad (4.5)$$

The input-voltage is in first order approximation solely dependent on the controlling current I_C, as could be expected. The influence of the input-current I_A is small, depending on the channel-length modulation effect $O(\lambda)$. The influence of the output-voltage V_B on the input-voltage V_A is extremely small, as it depends on the product of the modulation-parameters $O(\lambda^2)$. For a strong input-voltage stability

- the transconductances β_1 and β_2 should be large,
- and long transistors (small λ's) should be used.

4.2.2 Small Signal Stability

The current-conveyor circuit from figure 4.5(b) has a high loop gain, resulting in excellent large-signal voltage stability, as was shown in the previous section. Due to this high gain however, the circuit can easily become instable for small signals, as parasitic capacitances can decrease the phase-margin considerably. In this specific application with many multipliers connected to a single conveyor, a wide range input-range of currents is necessary, resulting in varying parasitic capacitances and varying impedances from the current-source at the input. Hence a careful analysis is necessary. For analysis, impedances of the different sources (at terminal A and C) and the load (at terminal B) have to be taken into account. The complete equivalent small signal circuit is shown in figure 4.6.

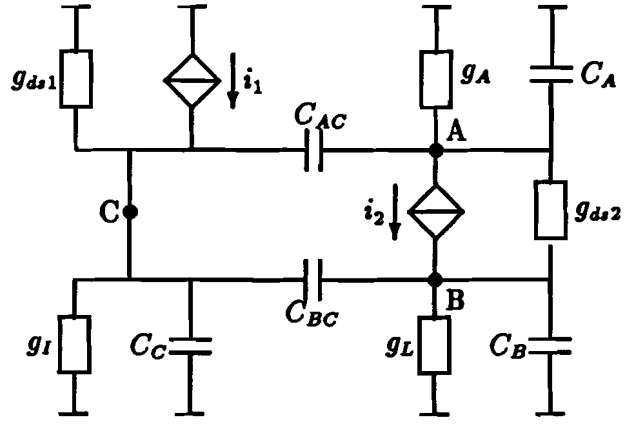

Figure 4.6: Equivalent small signal circuit for the current conveyor.

In this circuit, C_A, C_B and C_C represent the total parasitic capacitance to ground at node A, B and C respectively, represents g_I the conductance of the current source, g_A the total conductance of the connected multipliers, and g_L the conductance of the load at node B. Finally, C_{AC} is the sum of the gate-drain capacitance of M1 and the gate-source capacitance of M2, and C_{BC} equals the gate-drain capacitance of M2. To calculate the open-loop gain for stability analysis,

[1] All second- and higher-order distortions (λ^2) are neglected, unless a variable only has second- or higher-order influence. In this case only the lowest order distortion of this variable is taken into account.

the current in the voltage-controlled current source (VCCS) of transistor M1 is now determined by a fictional input voltage v'_A. The currents in both VCCS are

$$\begin{aligned} i_1 &= -g_{m1} \cdot v'_A \\ i_2 &= g_{m2} \cdot (v_A - v_C) \end{aligned}$$

The open-loop gain is now defined by

$$A_{OL} = \frac{v_A}{v'_A}$$

Due to the numerous parasitic capacitances in this circuit, this open-loop gain is complicated function. An exact evaluation of this transfer-function and precise determination of all poles and zeroes is therefore hardly possible. By making some assumptions towards the relative values of the variables, the poles and zeroes can be determined, and easily verified by looking at the small-signal circuit. The overall evaluation of the stability of the circuit will not be changed drastically by making these neglects. However, extra attention has to be payed to stability when simulating the circuit. To estimate the poles and zeroes the following assumptions are made: (1) Drain-source conductances are small compared to other conductances $g_{ds} \ll g_m, g_L, g_A$, and (2) Inter-node capacitances are small compared to the node-to-ground capacitances $C_{AC} \ll C_A$ and $C_{BC} \ll C_B, C_C$. Now, the poles and zeroes of the open-loop transfer-function can be approximated by

$$\begin{aligned} p_1 &\approx -\frac{g_L}{C_B+C_{BC}} \approx \frac{-g_L}{C_B} & z_1 &\approx -\frac{g_{m2}}{C_{AC}} \\ p_2 &\approx -\frac{g_{ds1}+g_I}{C_C+C_{AC}+C_{BC}} \approx -\frac{g_{ds1}+g_I}{C_C} & z_2 &\approx -\frac{g_L}{C_B+C_{BC}} \approx -\frac{-g_L}{C_B} \\ p_3 &\approx -\frac{g_{m2}+g_A}{C_A+C_{AC}} \approx -\frac{g_{m2}+g_A}{C_A} \end{aligned} \tag{4.6}$$

Finally, the DC open-loop gain is

$$A_{OL,DC} \approx -\frac{g_{m1}}{g_I + g_{ds1}} \cdot \frac{g_{m2}}{g_{m2} + g_A} \tag{4.7}$$

Although not exactly identical, the effects of zero z_2 and pole p_1 will cancel out, resulting in a system with effectively two poles and a single zero. Depending on the values of the two poles and zero, the phase-margin of the system can become too small, resulting in a potentially instable system.

First the dominating pole is determined because, as in any feedback circuit, stability or a larger phase-margin can be obtained by decreasing the dominating pole. As the drain-source conductances of transistors are small, p_2 is the dominating pole. Fortunately, this pole does not depend on the input-current I_A. The DC-current I_C through transistor M1 is constant, resulting in constant drain-source conductances g_{ds1} and g_I. To insure stability or to increase the phase-margin an extra capacitance can be connected to node C.

Whether an extra capacitance is needed or not, depends on the value of pole p_3 and zero z_1. If $|p_3|$ is much larger than the gain-bandwidth product $A_{LO,DC} \cdot |p_2|$ no compensation is needed. The same holds for the special case that the zero cancels the effects of the second pole. However, pole p_3 and zero z_1 are not constant. Depending on the input-current I_A the transconductance g_{m2} of transistor M2 can change drastically, especially when a wide input-range is used. The joint conductance of the connected multipliers g_A is also affected, but this change in value is neglegible, as the multiplier-transistors are in linear operation[2].

[2]The drain-source conductance of a transistor in linear mode, is nearly independent of the drain-source voltage, but does depend on the gate-source voltage.

Because of this changing pole-zero pair, it is difficult to use the criterion for a certain phase-margin in the design of current-conveyer, but it is possible to make a rough estimate of both the gain-bandwidth product and the second pole. The gain-bandwidth product is

$$|A_{OL,DC}| \cdot |p_2| \approx \frac{g_{m2}}{C_C + C_{AC} + C_{BC}} \cdot \frac{g_{m1}}{g_{m2} + g_A}$$

As both sums of capacitors $C_C + C_{AC} + C_{BC}$ and $C_A + C_{AC}$ will be in the same order of hundreds of fF, no problems concerning the stability of the circuit can be expected if $g_{m1} \ll g_{m2} + g_A$.

4.2.3 Design

In designing (or actually sizing) the current-conveyor some additional aspects have to be taken into consideration, although the main aspect remains the output-voltage V_A, which is identical to the drain-voltage of the multipliers. The following considerations have to be made:

Output-voltage: The output-voltage is identical to the multiplier drain-voltage: $V_A = 3.5V$. In order to minimize the current I_C through M1, the transconductance β_1 of this transistor should not be too large. For voltage-stability however, this value, together with the transconductance β_2 of transistor M2 should be as large as possible. Choosing $I_C \leq 5\mu A$ results in a transistor-sizing $\frac{W_1}{L1} \approx 1$.

Dynamic Range: For correct operation of the current-conveyor, transistor M2 should remain in saturation-mode, limiting the maximum drain-voltage V_B. For a maximum dynamic range of the load-voltage V_B the transconductance β_2 should be as large as possible:

$$V_C \leq V_A - \sqrt{\frac{2I_{A,max}}{\beta_2}}$$

Choosing $\frac{W_2}{L_2} = 20$ results in a rather large, but still reasonable sized transistor, as far as the actual implementation (layout) is concerned. Now, a maximum drain-voltage of $V_B = 2.5V$ is possible in the case 20 multipliers are connected.

Stability: To ensure the phase-margin of the conveyor is as large as possible, the small-signal transconductance g_{m1} of transistor M1 should be much less than the small-signal transconductance g_{m2} of transistor M2. This is done by reducing the control-current I_C, reducing the transconductance β_1 and enlarging β_2. Fortunately, the joint conductance of the parallel connected multipliers g_A is very large, also increasing the phase-margin.

Summary: For the current-conveyor, the following applies:

- Input Voltage: $V_A = 3.5V$.
- Maximum input current: $I_{A,max} = 120\mu A$.
- Maximum drain voltage: $V_{B,max} = 2.5V$.
- Control Current: $I_C = 3.6\mu A$.
- Transistor sizings: $\frac{W_1}{L_1} = \frac{16\mu m}{12\mu m}$ and $\frac{W_2}{L_2} = \frac{96\mu m}{4.8\mu m}$.

The total circuit, consisting of two current-conveyors and 19 multipliers is shown in figure 4.7. For each cell in the neighbourhood, there are two multipliers (for input and output), leaving a single multiplier for the threshold. Using the two current conveyors, all differential currents are now added, resulting in two currents I_L and I_R, the difference of which contains the total cell input-information $\sum A_i \cdot y_i + \sum B_i \cdot u_i + I_i$.

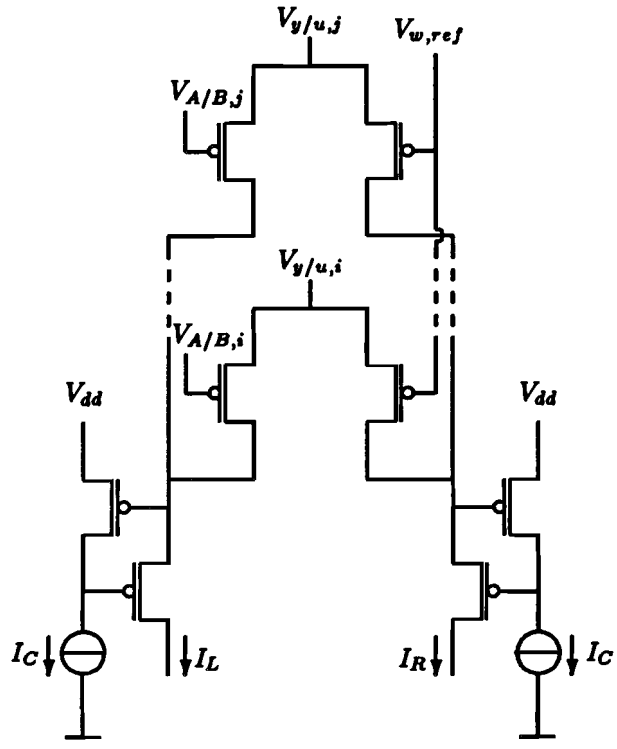

Figure 4.7: The total multiplier section of a single cell.

4.2.4 Simulations

For simulation of the current-conveyor, 20 single transistors of the 2T-multiplier were connected in parallel to the current input of the current-conveyor, with all gates connected to ground, resulting in a maximum current of 120μA. A linear resistor of 20kΩ was connected to the drain of the current-conveyor, resulting in a maximum drain-voltage V_B of approximately 2.5V. The common source of the multiplier-transistors was used as an input to control the input-current. The voltage-characteristics of the current-conveyor are shown in figure 4.8.

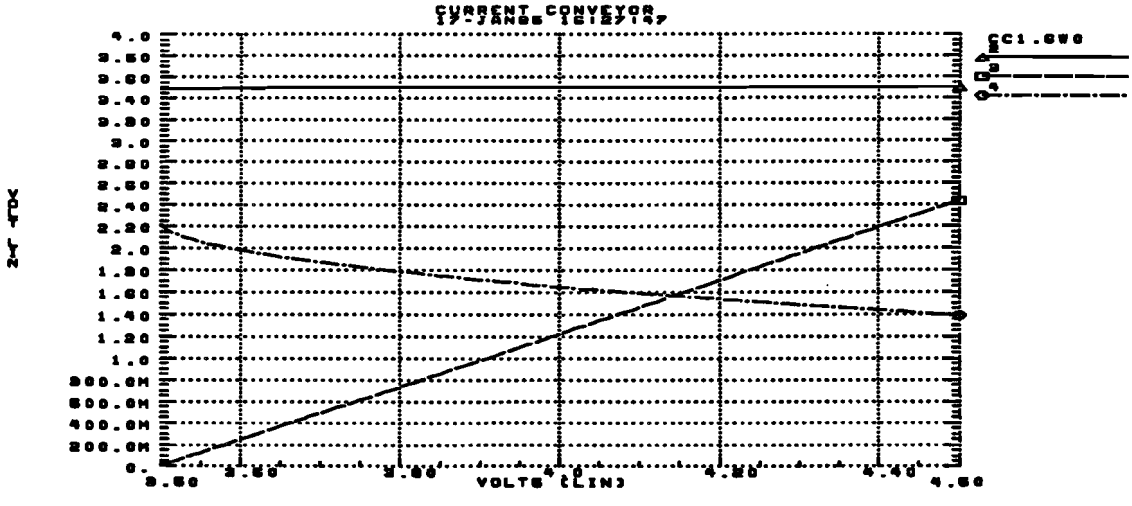

Figure 4.8: The voltage-characteristics of the current-conveyor.

In this figure V(2), V(3) and V(4) represent the input-voltage V_A, the drain-voltage V_B and the gate-voltage V_C respectively of the current conveyor in figure 4.5. The input-voltage V_A is almost constant throughout the entire current input-range. Closer examination learns that the maximum variation in the input-voltage is less than 10mV. Although the drain-voltage V_3 increases to approximately 1V above the gate-voltage V_4 Transistor M2 remains in saturation as the threshold-voltage is increased by the, non-zero, source-bulk voltage.

Using operation-point information generated at the previous simulation, the equivalent *small-signal* circuit is used to evaluate the open-loop gain. The values for the specific elements are:

g_{m1}	=	11.3μ	A/V	g_{ds1}	=	27.9p	A/V
g_{m2}	=	0...280μ	A/V	g_{ds2}	=	347p	A/V
g_A	=	120μ	A/V	C_A	=	1.2	pF
g_L	=	50μ	A/V	C_B	=	1	pF
g_I	=	43p	A/V	C_C	=	50	fF
C_{AC}	=	250	fF	C_{BC}	=	25	fF

The transfer-function for maximum transconductance g_{m2} is shown in figures 4.9 and 4.10.

Figure 4.9: The open-loop gain (magnitude) of the current-conveyor.

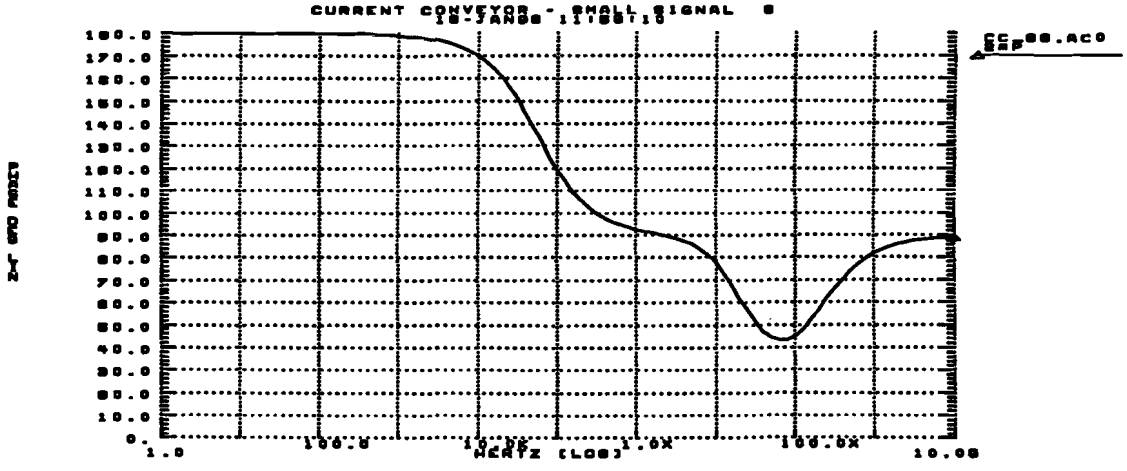

Figure 4.10: The open-loop gain (phase) of the current-conveyor.

Looking at the bode-plot in figure 4.10 two poles and a single zero can be clearly recognized.

As the secondary pole and zero are close together, both lying well above the unity-gain bandwidth, the phase-margin is very large. As a result the circuit will be stable, acting as a low-pass filter with a single pole. Using the values for the different elements this dominant pole can be calculated: $p_2 \approx 20$kHz. This value can be verified by the bode-plot.

4.3 Conclusions

When implementing a multiplier for operation in two quadrants, a two-transistor multiplier together with two current-conveyors can be used. In the case the outputs of many multipliers have to be added for further processing, a very common feature for neural nets, the two conveyors form only a marginal overhead on the total implementation area of the multipliers.

Using two small (W=4.8μm, L=27.2μm) PMOS transistors, large input- and output-ranges (2V, 1V and 1.7μA respectively) can be obtained for the multiplier. Although the multiplier is inherently linear, secondary effects lead to some distortions. One of the main effects, which leads to the non-linearity in the bottom curves in figure 4.2 is the mobility-reduction due to surface-scattering at high gate-source voltages. However, this effect strongly depends on the model and model-parameters used in the simulations. Comparing the relevant parameters (UCRIT, UEXP) to parameters of similar processes shows that these values are somewhat exaggerated. The amount of non-linearity as shown in figure 4.2 therefore becomes questionable and should be verified by measurements.

Two current-conveyors are used to keep the drain-voltages of the multipliers at a constant and equal level. The main distortions in this operation are due to the limited loop-gain and the channel-length modulation effect. Using a very wide, and short (but not minimum-length) transistor these effects become very small, and the deviation in input-voltage (or multiplier drain-voltage) becomes less than 10mV for the total input-range of 120μA for each conveyor. The *difference* in multiplier drain-voltages is even smaller, as the currents in both conveyors have approximately the same bias, and can differ no more than 40μA. The main error in multiplier drain-voltages will therefore be a result of zero-bias threshold-voltage deviations in transistor M1 of the conveyor, forming an offset to the differential current. This offset depends on the gate-voltages of the multipliers, and therefore changes with the weights used.

The small-signal stability of the current-conveyor mainly depends on the output-impedance of the input-current source and the transconductance of the two transistors. Stability can be improved by a small output-resistance of the multipliers and a small ratio of transconductances β_1/β_2. As many multipliers are connected in parallel, giving a small output-resistance, and transistor M1 has a small drain-current and a small transconductance compared to transistor M2, the conveyor is stable, having a phase-margin of more than 90°.

The total multiplier-section output consists of two currents, the difference of which contains the total cell-input information. The maximum current in each output-terminal is less than 120μA, and the maximum difference in currents does not exceed $20 \cdot 2\mu\text{A} = 40\mu\text{A}$. The voltage at each output-terminal should not exceed 2.5V to insure correct operation of the current-conveyor. Using a power supply of 5V and having a maximum multiplier current of 10μA, the total power consumption of the multiplier section of a single cell is less than 1mW.

Chapter 5

Cell Core Design

To implement the cell-dynamics and the output-function of a CNN-cell, the so-called cell-core has to be designed. In this part of the cell, the total cell-input $\sum A_i \cdot y_i + \sum B_i \cdot u_i + I_i$ has to be fed to a lossy integrator resulting in a state x_i, from which the output y_i is determined using a non-linear function. The design of this cell-core does not depend on the choice of the multiplier used, as both multipliers presented in 3.3 have (differential) output currents.

5.1 State Implementation

To implement the cell-dynamics, or degrading cell-memory, the total cell-input is fed to a lossy integrator. As the input is a current, this integration can be easily done using a capacitor-resistor pair, as is shown in the original implementation of Chua and Yang (see figure 2.4). As in this case the input consists of two currents, the difference of which contains the actual cell input, a slight modification is needed. This results in the differential equivalent of the capacitor-resistor pair, as shown in figure 5.1.

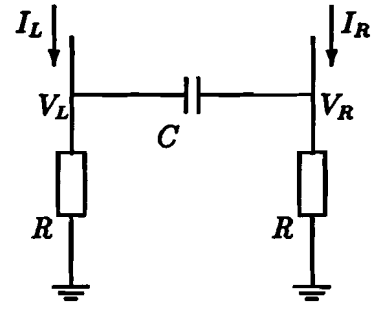

Figure 5.1: Differential equivalent of a capacitor-resistor pair.

Using this circuit, the state is represented by the differential voltage $V_x = V_L - V_R$. The dynamic equation for this sub-circuit is

$$2C \cdot \frac{d(V_L - V_R)}{dt} = -\frac{1}{R}(V_L - V_R) + (I_L - I_R) \tag{5.1}$$

This configuration brings along some problems, also present in the implementation of Chua and Yang. The first problem is the *dynamic range* of the state-voltage V_x, already explained in chapter 2. If the template-elements are large, the unit state-voltage is very small, as the state-voltage should not be limited by the physical implementation. This results in poor accuracy of the state-voltage, and thus of the output-voltage of the cell. In this specific configuration, the maximum state-voltage $V_{x,max}$ is determined by the ranges of V_L and V_R. When using the 2T-multiplier, this voltage equals the maximum output voltage of the current-conveyor. Due to the rather large bias which can be present in the multiplier-currents I_L and I_R, the maximum voltage-range of the state V_x, and thus the unit state-voltage are reduced considerably (as the resistors R are limited by these bias-currents). Implementation of the Full-Range principle by limiting the state-voltage also becomes very difficult, as a differential voltage representation of the state is used.

Another problem is the *transformation* to the output-voltage V_y. The range of the output-voltage is, of course, identical to the input-range of the multipliers, as the output is connected to the multipliers of all cells of a neighbourhood. The relation between the input of a multiplier and the output-voltage of the cell (identical to a template-element), is determined by several variables: the transconductance $2\beta_m \cdot (V_w - V_{w,ref})$ of the multiplier, the resistors R and the amplification A_{NL} of the non-linearity section. Obviously, this results in an indirect relation between the the desired template-element TE (which can be A_i or B_i or I_i) and the multiplier-voltage needed to obtain this element

$$TE = A_{NL} \cdot R \cdot \beta_m (V_w - V_{w,ref})$$

As this relation depends on several variables, it is more susceptable to distortions and inaccuracies, than would be case when a direct relation would exist between multiplier input and cell-output. This direct relation can be obtained by using a multiplier in feedback configuration to determine the state of the cell, a method also used by Nossek and Seiler [5] and modified by Perfetti [7] for Full-Range purposes.

5.2 State Feedback Implementation

To determine the input of a certain system in general, a feedback loop can be used: the output of a difference-amplifier is added to the original input, exactly compensating this original input. The output of the amplifier will therefore be identical, but sign-reversed, to the input of the system. In this specific case, the voltage-inputs of the system are connected to multipliers, resulting in a total input-current ΔI. To compensate for this current, another multiplier can be used, which is controlled by the output of a difference-amplifier, or op-amp. The output of the op-amp should now equal the sum of inputs of the connected multipliers, having the opposite sign. To correct this sign, the feedback-multiplier should be reversely connected. Although the relation between the input ΔI and the state V_x (output of the op-amp) is now direct, this feedback configuration brings along some problems.

For correct operation, the range of the feedback-multiplier should equal the total range of the cell-input $\Delta I = \sum A_i y_i + \sum B_i u_i + I_i$. Using 19 multipliers for the input, the feedback-multiplier should consist of another 19 multipliers connected in parallel, effectively dividing the cell-state V_x by 19. To determine the output V_y, an extra amplifier is needed. This problem can be solved by changing the CNN into a Full-Range CNN, as shown by Perfetti [7]. Now, a single multiplier in the feedback-loop is used, effectively limiting the state V_x, and making the amplifier superfluous.

Another problem is the implementation of the dynamics. As mentioned, the feedback-loop is merely used to determine the *input* ΔI, thereby not implementing the dynamics (or integration) of the state V_x. By connecting a capacitor in parallel with the feedback multiplier, the lossy integrator is implemented, as the feedback multiplier acts as a conductance.

The main problem arises however, when using this method in a two-quadrant CNN. The total input can, due to the template-elements, become both positive and negative. If a feedback-multiplier is used (with a certain weight), only one quadrant can be compensated. By adding an negative offset, the input *can* be compensated in two quadrants. This offset can be created by adding an extra multiplier using a fixed input-voltage and opposite weight, compared to the feedback multiplier.

Combining these requirements, a feedback configuration can be designed for the cell core, implementing the basic dynamics of a Full-Range CNN. This cell core is shown in figure 5.2 for the implementation using the 2T-multiplier. A similar circuit can be designed to be used

with the differential stage multiplier by simply exchanging the multipliers, and leaving out the current-conveyors.

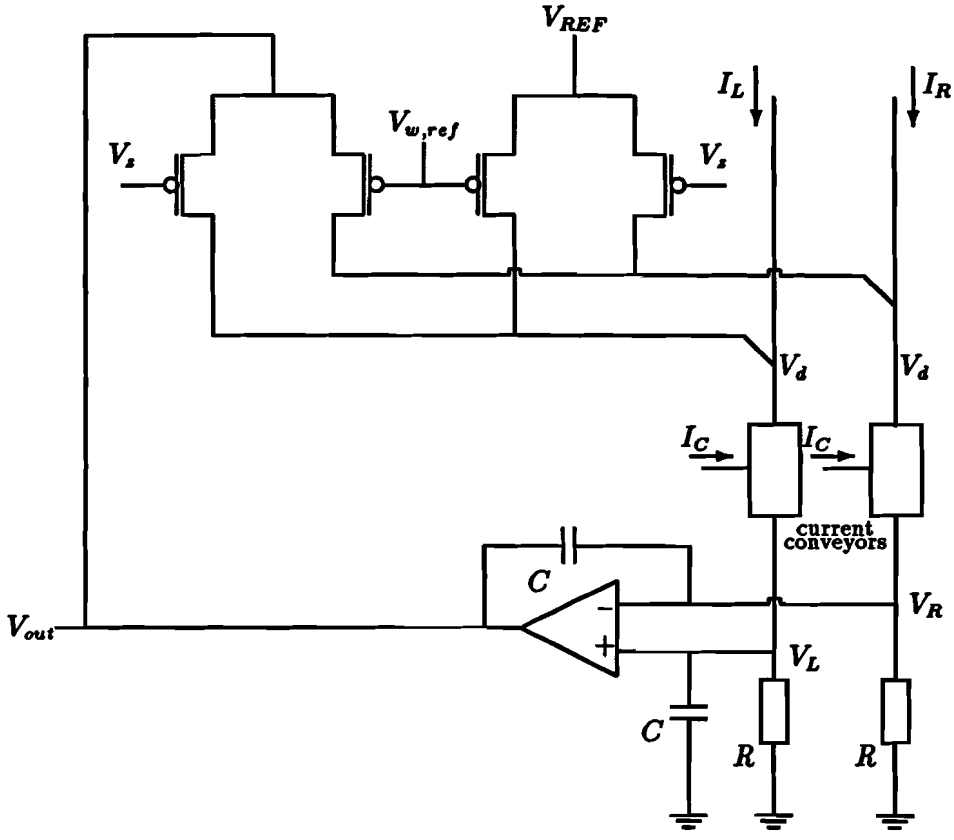

Figure 5.2: Cell Core of a Full Range CNN.

For correct operation, the output of the op-amp V_{out} should be limited or *clipped* to the voltage-input range of the multiplier. For the 2T-multiplier this range is 3.5V to 4.5V. Using a non-ideal op-amp with amplification A and a dominant time-constant τ_{OA}, the dynamic equation for this sub-circuit, in non-clipped mode, can be derived (see Appendix C)

$$\left[C + \frac{\tau_{OA}}{AR}\right] \cdot \frac{d(V_{out} - V_{REF})}{dt} = -\beta_m(V_{out} - V_{REF})(V_z - V_{w,ref}) + I_L - I_R \qquad (5.2)$$

By writing $I_L - I_R$ as the sum of outputs of the multipliers, this equation can be rewritten as

$$\left[\frac{C + \tau_{OA}/AR}{\beta_m(V_z - V_{w,ref})}\right] \cdot \frac{dV_{out}}{dt} = -(V_{out} - V_{REF}) + \sum_i \frac{(V_{w,i} - V_{w,ref})}{(V_z - V_{w,ref})} \cdot (V_{in,i} - V_d) \qquad (5.3)$$

From this dynamic equation, the following can be concluded:

1. In this configuration $V_z = V_{out} - V_{REF}$ represents the state x of the cell, as this voltage is proportional to the input ΔI (in steady state). The output of the cell is represented by $V_y = V_{out} - V_d$, as this is the actual voltage that is multiplied in all connected multipliers, including the feedback-multiplier. If V_{REF} is chosen exactly in the middle of the output-range of V_{out}, $V_{REF} = 4V$, the state x operates symmetrically in two quadrants, and the output y operates in a single quadrant. This voltage V_{REF} is now identical to the reference

voltage needed for edge-cells, as discussed in section 3.2. Important to note is that the ratio between the state-voltage V_x and the output-voltage V_y is always *unity* in the non-clipped region. With respect to the two-quadrant non-linearity function this implies: $-\frac{1}{2} \leq x \leq \frac{1}{2}$ and $0 \leq y \leq 1$.

2. As the op-amp output-voltage V_{out} is limited by $V_{y,min} = V_d$ and $V_{y,max}$ the state of the cell is also limited. This cell-core thus implements a Full-Range CNN.

3. The time-constant of the cell depends on the capacitors C, the gain-bandwidth product of the op-amp $GB_{OA} = A/\tau_{OA}$, the resistors R and the transconductance $\beta_m(V_x - V_{w,ref})$ of the feedback-multipliers.

$$\tau_{cell} = \frac{C + \tau_{OA}/AR}{\beta_m(V_x - V_{w,ref})} \tag{5.4}$$

As the capacitors only increase the time-constant, and do not contribute to the essential operation of this sub-circuit, the capacitors could be left out. However, small capacitors will be necessary to store the initial state. If small enough, the time-constant of the cell will mainly depend on the gain-bandwidth of the op-amp. Furthermore, the time-constant depends on the weight-voltage of the feedback-multiplier $(V_x - V_{w,ref})$, which is used to determine the template-elements. As a result, this voltage should be identical for every cell in the net, as all cells need to have an equal time-constant.

4. The template-elements are determined by the ratio between the corresponding weight-voltage $(V_{w,i} - V_{w,ref})$ and the feedback weight-voltage $(V_x - V_{w,ref})$. As the latter voltage must be identical for every cell, it can be regarded as a template-scaling input. This voltage however does *not* change the ratio between the state-voltage V_x and the output-voltage V_y. Finally, to ensure negative feedback the control-voltage V_x should be larger than the weight-reference voltage: $V_x > V_{w,ref}$.

5.3 Op-Amp Design

An important element in the circuit of the cell-core is the op-amp. For this op-amp a special design has to be made, as the output should have a limited or clipped output-range. The main part of the circuit will however consist of common amplifier circuitry, to which some general requirements will apply. First, a common op-amp satisfying these requirements will be designed, which will be modified afterwards to perform a clipping function.

5.3.1 Requirements

Apart from the usual requirements such as stability, low power consumption and a high gain-bandwidth product, this specific application requires some extra properties concerning input- and output-ranges, slew-rate and gain-bandwidth product.

Input-Range: As both inputs of the op-amp are connected to a resistor and the output of a current-conveyor, the input-range is determined by the range of the voltages V_L and V_R. As said earlier, the maximum of these voltages is limited by the current-conveyor and equals 2.5V. The input-values of the op-amp thus lie between 0V and 2.5V, and a PMOS input-stage for the op-amp is required. This maximum input-value also determines the value of resistor R. As mentioned in the previous chapter, the current through the current-conveyor does not exceed 120μA. The maximum value of the resistor therefore equals $R = 20.8\text{k}\Omega$.

Output-Range: The output-range of the op-amp is, as can be seen in figure 5.2, equal to the input-range of the 2T-multiplier. The output thus lies between 3.5V and 4.5V and should be clipped to these values. The output of the op-amp is not only connected to the feedback-multiplier, but also to template-multipliers of all cells the neighbourhood, resulting in a total of ten connected multipliers. If the output is high (4.5V) all multipliers will drain approximately 8 to 10μA, resulting in a total output-current of at most 100μA. If the output is low (3.5V) no current will flow through the multipliers, and the total output-current will be zero. Using a PMOS inverter with current-source, the bias-current in the output-stage of the op-amp could be small, as no current needs to be drained from the multipliers.

Gain-Bandwidth: The gain-bandwidth product is defined by:

$$GB_{OA} = \frac{A_{OA,DC}}{\tau_{OA}} \tag{5.5}$$

The time-constant of the cell will mainly depend on the gain-bandwidth product of the op-amp. As this time-constant should be identical for each cell in the net, it should not depend on parasitic elements, which could vary from cell to cell. If a so-called Miller-capacitance is used to ensure stability of the amplifier, the gain-bandwidth of the cell is determined by this capacitor. As matching between capacitors mainly depends on sizing, fluctuations in the gain-bandwidth are likely to be decreased.

The DC-gain of the op-amp should be large to minimize the steady-state errors in the output of the cell. For a total error of less than 1% in the output, the total DC-loop-gain should be more than 100. The total loop-gain LG of the sub-circuit equals

$$LG_{DC} = A_{OA,DC} \cdot \beta_m R(V_z - V_{w,ref})$$

Assuming that $(V_z - V_{w,ref}) > 0.1V$, resulting in maximum template-elements of 10, the DC-gain of the op-amp should be larger than $1000/R\beta_m = 16 \cdot 10^3$, or 84 dB.

Slew-Rate: The required slew-rate of the op-amp can be obtained from the differential cell-equation (5.2). As the total current $I_L - I_R$ can become relatively large compared to feedback-term on the right-hand side of the equation, it will predominantly determine the slew-rate. The maximum slew-rate is

$$SR_{max} = \left.\frac{\beta_m(V_z - V_{w,ref})(V_{out} - V_{REF}) + I_L - I_R}{C + \tau_{OA}/AR}\right|_{max} \tag{5.6}$$

The total current $I_L - I_R$ equals the sum of difference-currents of 19 multipliers:

$$I_L - I_R = \sum_{i=1}^{19} \cdot \beta_m(V_{w,i} - V_{w,ref})(V_{in,i} - V_d)$$

Substituting this current, and substituting the maximum values for all differential voltages and $V_{in,i} - V_d$, the maximum slew-rate needed for the op-amp can be determined:

$$SR_{max} \approx 0.81 \cdot GB_{OA} \qquad V/s$$

This restriction will predominantly determine the maximum gain-bandwidth and corresponding slew-rate, as gain and slew-rate can be exchanged while maintaining stability and input- and output-ranges. This trade-off will be treated in the next section.

5.3.2 Design

To design an op-amp, three important steps have be taken. First, all the requirements for the op-amp have to be determined. This has already been done in the previous section. Secondly, a suitable topology or structure has to be chosen, depending on special requirements, e.g. input-range and output-load. Finally, all the circuit-elements (mainly transistors) have to be sized, according to the required properties, to achieve an as large as possible gain-bandwidth product.

Structure

In chosing a suitable structure for the op-amp, properties such as output-load, gain and input-range play an important role. The first or *input-stage* of an op-amp, usually consists of a differential pair with a certain load. The common-mode input-range of the op-amp equals, as mentioned, 0V to 2.5V. Therefore, a PMOS differential pair should be used, as NMOS transistors would be cut off using such a low gate-voltage. As the gate-voltages of these transistors can become as low as 0V, and the transistors should remain in saturation for optimal operation, a cascode load-circuit can be used (see [9]): Using two bias-currents, the signal currents are mirrored to a PMOS current-mirror. Two cascode-transistors with fixed gate-voltages V_{cas} keep the drain-voltages of the differential pair transistors at a constant low voltage to ensure saturated operation, and create a large loading impedance for the first stage. This circuit, comprising transistors M0... M4 is shown in the figure 5.3.

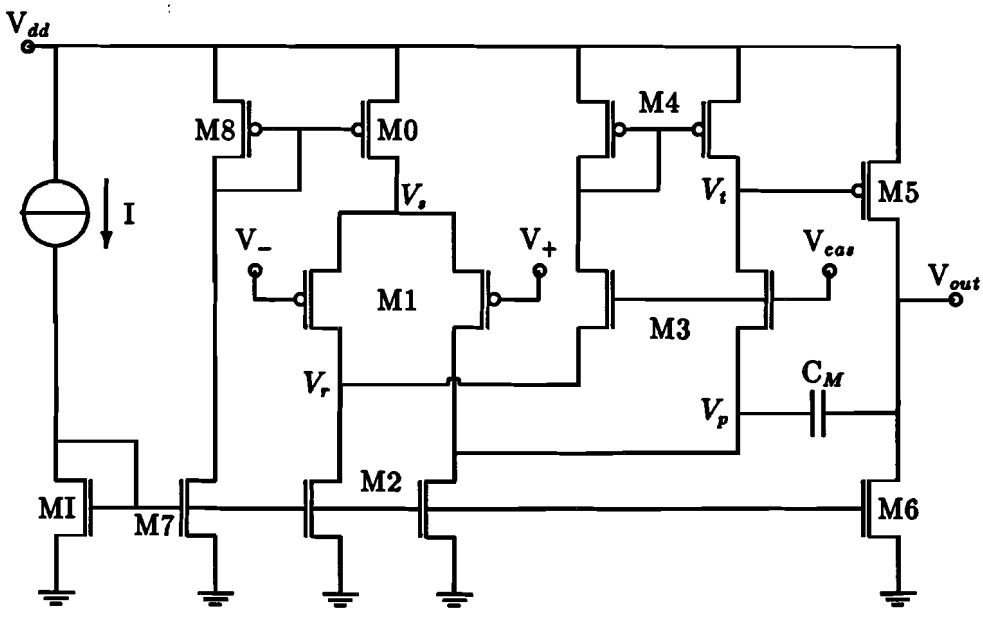

Figure 5.3: A two-stage cascode op-amp with miller compensation.

The *second stage* of the op-amp could consist of an intermediate amplifier section, but might also serve as the output-stage of the op-amp. In order to have high gain, an inverter-stage can be used. As the input of the second stage is determined by a PMOS current-mirror, a PMOS inverter using an NMOS current-source will be used. This circuit comprises transistors M5 and M6 in figure 5.3.

To ensure *stability* and fix the *gain-bandwidth* independently from parasitic influences, a so-called Miller feedback capacitor is inserted between the source of the appropriate cascode-

transistor and the output of the second stage. Usually, a miller-capacitor is connected between the output of the second stage and the high-impedant output of the first stage, to create a dominant pole. Although the capacitor is now connected to smaller impedance, the effective capacitor is proportionally higher, as the gain from the source of the cascode to the output is larger, thus creating the same dominant pole.

Properties

For the circuit shown in figure 5.3, the requirements stated in section 5.3.1 can be translated into restrictions for the corresponding properties of the op-amp. From the total set of possible solutions satisfying all constraints an optimal solution can then be chosen. In order to simplify the design some assumptions concerning currents and transistor sizings will be made: transistors MI, M2 and M7 will be identical, and transistors M0 and M8 will also be identically sized. This will result in equal drain-currents of transistors M0 and M2.

The common-mode *input-range* of the op-amp is determined by the input-stage. For correct operation, all transistors M0...M2 should remain in saturation. To keep transistors M1 into saturation, the drain-voltage V_r should be lower than the threshold-voltage $|V_{T1}|$:

$$V_r \leq |V_{T1}| \tag{5.7}$$

To keep transistors M2 into saturation, the gate-voltage should be not too high, or put in another way, sufficient current ($I_{d2} = I$) has to flow through transistors M2 while remaining in saturation with a low drain-voltage V_r:

$$\sqrt{\frac{2I}{\beta_2}} \leq V_r \tag{5.8}$$

Finally, transistor M0 should remain in saturation in order to retain the operation as a current-source. The drain-voltage, depending on the gate-voltages of transistors M1, is highest if these gate-voltages are identical, both equal to the maximum input-voltage $V_{in,max}$.

$$V_s = V_{in,max} + |V_{T1}| + \sqrt{\frac{I}{\beta_1}} \leq V_{dd} - \sqrt{\frac{2I}{\beta_0}} \tag{5.9}$$

As a considerable *output-current* is needed, output-transistor M5 should be large enough to be able to source this current without the output-voltage dropping below the maximum output-voltage $V_{out,max}$. Transistor M5 will be operating in linear mode when such a high current (and output-voltage) is needed, and gate-voltage V_t will approximate V_r:

$$V_{sd} = V_{dd} - V_r - |V_{T5}| - \sqrt{(V_{dd} - V_r - |V_{T5}|)^2 - \frac{2(I_{out} + I_{d6})}{\beta_5}} \leq V_{dd} - V_{out,max}$$

resulting in

$$V_r + |V_{T5}| + \sqrt{(V_{dd} - V_r - |V_{T5}|)^2 - \frac{2(I_{out} + I_{d6})}{\beta_5}} \geq V_{out,max} \tag{5.10}$$

The *slew-rate* of the op-amp will mainly be determined by the charging and dis-charging of parasitic and loading capacitances and the Miller capacitance C_M. The latter capacitance will predominantly determine the slew-rate, as parasitic capacitances will be relatively small and the loading of the op-amp will be high (small impedance), thus reducing the effect of the loading

capacitance. The slew-rate is now determined by the Miller-capacitor and the minimum current possible flowing through the capacitor, which is either I or I_{d6}.

$$SR_{oa} = \frac{\min(I, I_{d6})}{C_M} \tag{5.11}$$

To evaluate the *small-signal behaviour* of the op-amp, including stability and gain, an equivalent small-signal circuit has to be derived. As usual, this circuit consists of voltage-controlled current-sources (VCCS) and impedances, and is shown in figure 5.4.

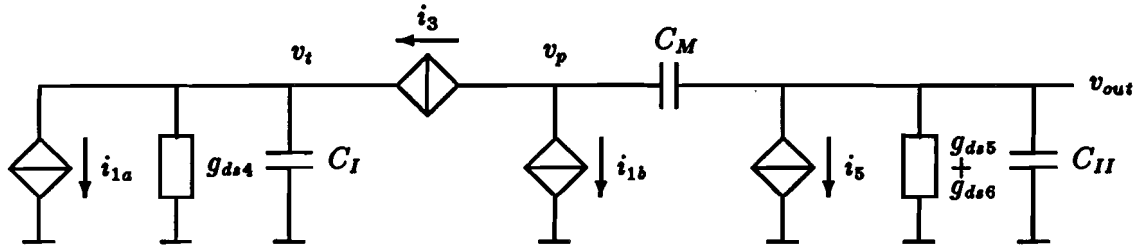

Figure 5.4: Equivalent small signal circuit of figure 5.3

The VCCS of the first stage is formed by transistors M1, having a transconductance g_{m1}. This results in a current

$$i_{1a} = i_{1b} = g_{m1}(v_+ - v_-)/2$$

for both VCCS. Due to the cascode-configuration, transistors M2 appear to have no conductance, and current-mirror M4 effectively forms the load-circuit for the left-hand VCCS of M1 (g_{m1}). The load therefore consists of the conductance g_{ds4} in parallel with the parasitic capacitance C_I at output-node v_i of the first stage. The right-hand VCCS of M1 is connected to the same load through the cascode VCCS of M3 (g_{m3}), having a current equal to

$$i_3 = g_{m3}v_p$$

Transistor M5, having a transconductance g_{m5} forms the VCCS of the second stage, which is loaded by the conductances g_{ds5} and g_{ds6} in parallel to the parasitic capacitance C_{II} at the output of the op-amp. The current resulting from the VCCS is

$$i_5 = g_{m5}v_i$$

Finally, a Miller-capacitor C_M is connected to the output of the second stage and the low-impedant source of the cascode-transistor M3. This configuration effectively creates a large capacitive load $A_{p,o} \cdot C_M$ for the first stage, in which $A_{p,o}$ denotes the voltage-gain of the source v_p of the cascode-transistor to the output v_{out} of second stage. Together with the source-impedance of the cascode-transistor this effective capacitance results in a dominant pole:

$$p_1 = -\frac{g_{m3}}{A_{p,o}C_M} = -\frac{g_{m3}g_{ds4}(g_{ds5} + g_{ds6})}{g_{m3}g_{m5}C_M} = -\frac{g_{ds4}(g_{ds5} + g_{ds6})}{g_{m5}C_M} \tag{5.12}$$

To determine the other poles and zeroes the total transfer-function should be determined. By neglecting minor terms this transfer-function can be approximated by:

$$H(s) \approx \frac{-s^2\frac{g_{m1}}{2}C_MC_I + s\frac{g_{m1}g_{m3}}{2}C_M + g_{m1}g_{m3}g_{m5}}{s^3C_{II}C_MC_I + s^2g_{m3}(C_{II} + C_M)C_I + sg_{m3}g_{m5}C_M + g_{m3}g_{ds4}(g_{ds5} + g_{ds6})} \tag{5.13}$$

The sign-sequence in the polynomial of the numerator can only result from two real zeroes, one in the left half (LHP) of the complex plane, and one in the right half plane (RHP). Although the latter zero causes a negative phase-shift which can decrease the phase-margin, no compensation through the use of a nulling-resistor (connected in series with the Miller-capacitor) is necessary, as the zero is very high, resulting from parasitic capacitances parallel to a low impedance (a transconductance of a VCCS).

Apart from the dominant pole, a conjugated pair of poles is present in the LHP. Such a conjugated pair can best be represented by it's real part, or actual (natural) frequency, and the quality-factor Q

$$\mathrm{Re}\{p_{2,3}\} \approx -\sqrt{\frac{g_{m3}g_{m5}}{C_I C_{II}}} \tag{5.14}$$

$$Q \approx \sqrt{\frac{g_{m5}C_{II}}{g_{m3}C_I}}\frac{C_M}{C_M + C_{II}} \tag{5.15}$$

To establish a phase-margin of at least 60° and to ensure stability, the real part of the conjugated pair of poles $p_{2,3}$ has to be approximately three times higher than the gain-bandwidth product of the op-amp. This results in a minimum value for the Miller-capacitor, as it determines the gain-bandwidth.

$$C_M \geq 3g_{m1}\left(\frac{C_I C_{II}}{g_{m3}g_{m5}}\right)^{1/2} \tag{5.16}$$

Furthermore, the quality-factor Q has to be sufficiently low to avoid a resonance-peak in the gain of the op-amp, which can decrease the *gain-margin* dramatically. As the poles will be placed near the unity-gain bandwidth, a small resonance-peak can increase the gain above unity again. This can result in a gain > 0 dB for a phase-shift of 180°.

Two other important aspects can also be easily derived from the transfer-function (5.13). The DC-gain is found by substituting $s = 0$, and the result can be easily verified by looking at figure 5.4.

$$A_{OA,DC} = \frac{g_{m1}g_{m5}}{g_{ds4}(g_{ds5} + g_{ds6})} \tag{5.17}$$

The gain-bandwidth product can now be calculated by multiplying the DC-gain and the dominant pole p_1:

$$GB_{OA} = \frac{g_{m1}}{C_M} \tag{5.18}$$

Sizing

Using the requirements from section 5.3.1 and the properties described in this section, a suitable sizing for the elements of the op-amp can be made. Although no optimalization criterion can be used to calculate all sizings *exactly*, some rough guidelines can be extracted from requirements and properties, to achieve an as large as possible gain-bandwidth product while retaining a high slew-rate.

1. For a large gain-bandwidth product and a large slew-rate, the Miller-capacitor should be as small as possible. This can be achieved by
   - decreasing the ratio $g_{m1}/\sqrt{g_{m3}g_{m5}}$
   - and minimizing parasitic capacitances by using small transistors.
2. Decreasing the transconductance g_{m1} by decreasing β_1, results in

- an unchanged gain-bandwidth, as the Miller-capacitor C_M can be smaller, proportional to g_{m1}, as is shown in (5.16),
- a larger slew-rate, as C_M can be smaller,
- and a smaller common-mode input-range, when maintaining the same current I.

3. Increasing the current I results in
   - a larger slew-rate,
   - a larger gain-bandwidth, as g_{m1} is proportional to the square root of I,
   - and a smaller common-mode input-range.
4. Increasing the transconductance g_{m5} or increasing g_{m3} results in
   - a higher gain-bandwidth product,
   - and a higher slew-rate, as C_M can be smaller.

As can be concluded from these guidelines, an important trade-off exists between the current I and the transconductance β_1 of transistors M1. To achieve the high ratio needed between the gain-bandwidth and the slew-rate of the op-amp, the transconductance β_1 of transistors M1 should be small to allow for a relatively high current I, while retaining the required common-mode input-range. This, of course, reduces the gain-bandwidth product. Now, to reduce the Miller-capacitance C_M and increase the gain-bandwidth and slew-rate simultaneously, a large transconductance β_5 and a large cascode g_{m3} has to be chosen. As a result, the gain of the second stage will be large, resulting in a effectively larger capacitive load for the first stage, thus allowing a reduction of C_M.

Summary

Now, using the requirements for the op-amp and following these guidelines, suitable sizings for the transistors can chosen. The final element-values are shown in table 5.1.

Table 5.1: Final element-values for the op-amp.

transistor	Width (μm)	Length (μm)
M0	24	4.8
M1	4.8	5.6
M2	12	12
M3	48	4.8
M4	24	12
M5	48	4.8
M6	24	12
M7	12	12
M8	24	4.8
Variable	Value	
C_M	0.4 pF	
I	10 μA	
V_{cas}	2.0 V	
V_{dd}	5.0 V	

These transistor-sizings and element-values result in the following properties:

- The common-mode input-range: $V_{cm} \geq 3.5V$.
- The slew-rate: $SR_{OA} \approx 13\ V/\mu s$.
- Power dissipation: $P_{OA} \leq 350\ \mu W$.
- Small-signal Gain: $A_{OA} \geq 85$ dB.
- Gain-Bandwidth Product: $GB_{OA} \approx 3.5$ MHz.
- Phase-Margin: $\phi_{OA} \geq 60°$.

5.3.3 Clipping

In order to establish the non-linearity of the Full Range cell, the state and the output of the cell should be limited. As both variables depend on the output-voltage of the op-amp (see section 5.2), limiting (or clipping) this voltage would create the desired non-linearity. For the implementation of the clipping function three common methods exist:

1. Using an extra stage between the output of the op-amp and the input of the feedback-multiplier (=output of the cell). This extra stage could make use of the voltage-limiting properties of diodes, as is shown in figure 5.5(a), or the current-limiting properties of a differential stage, as depicted in figure 5.5(b).

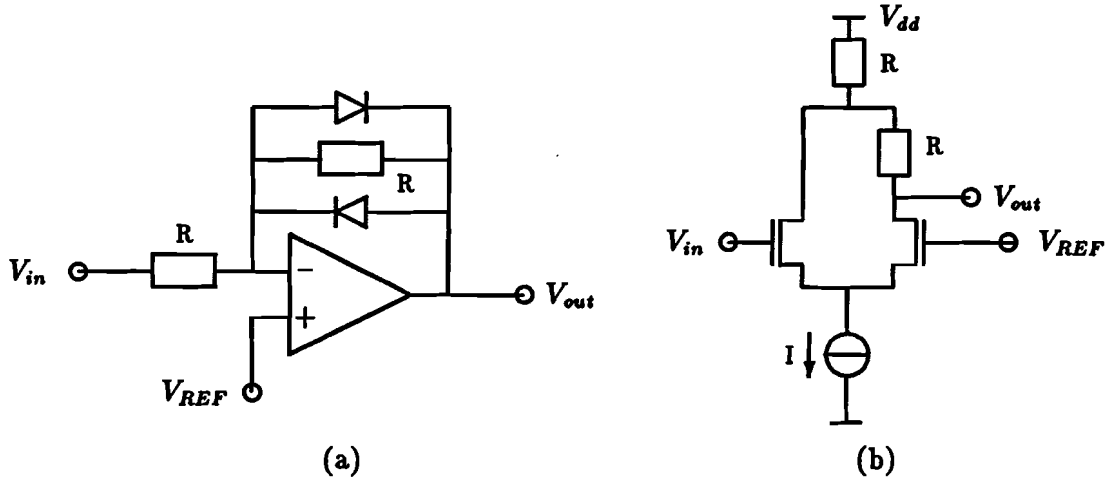

Figure 5.5: Clipping of the output of an op-amp using an extra stage.

Both solutions however suffer from some deficiencies. Using the first option, a unity-gain follower with two diodes connected anti-parallel between the output and a virtual reference node, the output-range cannot be controlled, but directly depends on the bandgap-voltage of the diodes. The output-range of the latter option, using a differential stage, is in this specific case determined by the current I and the resistors R

$$V_{dd} - IR \leq V_{out} \leq V_{dd} - 2IR$$

but other methods might be used to convert the differential current into an output-voltage. The main problem of this circuit however consists of the poorly defined input-range, which depends on absolute accuracies of the current I and transistor-sizings.

2. Connecting the two output-transistors M5 and M6 of the op-amp (see figure 5.3) to the desired clipping voltages. In this way, the output-voltage is hard limited to the upper clipping voltage $V_{hi} = 4.5V$ or the lower clipping voltage $V_{lo} = 3.5V$. A major drawback of this circuit is the output-current. Due to the loading of the output-stage by the connected multipliers, a considerable voltage-drop will occur across the PMOS output-transistor M5, depending on the loading-current. Moreover, as two extra voltages are needed in each cell, extra connections and thus extra implementation area are required.

3. Reducing the amplification. The output of the op-amp is compared with the fixed reference- or clipping-voltages. If the output of the op-amp exceeds these boundaries, the amplification is reduced. This reduction can be achieved by, for example, reducing the current of a differential input-stage. However, this method can only be used for both boundaries if the output-bias-voltage, i.e. the output-voltage with zero amplification, lies within the desired operating range. Usually, this will not be the case, and this method can only be used for a single boundary. However, when using this method for an op-amp in a feedback configuration, stability problems will occur when the output is near a clipping-voltage. In this case the clipping-circuitry will start to take effect on the behaviour of the op-amp. As a result, two feedback mechanisms will be present having opposite operations, creating a potentially instable circuit which is difficult to analyse.

Although no complete solution for the implementation of the non-linearity can be derived from these methods, a partial solution for limiting the output-voltage of the op-amp to the lower boundary is designed, based on the hard voltage-limiters of the second method and the specific requirements of the configuration in which the op-amp is used.

As the desired ouput-range of the op-amp is equal to the input-range of the multipliers, the voltage limitation used for these multipliers, i.e. the current-conveyor, could also be used for the output-stage of the op-amp. If the source of the lower output-transistor M6 of the op-amp is connected to the fixed voltage of the conveyor, an internal voltage-source is created. This modification in the output-stage of the op-amp is shown in figure 5.6. In order to be able to use this circuit some requirements/properties apply:

- As the threshold-voltage of transistor M6′ is increased due to the bulk-effect, the transistor will only operate in linear region when the output-voltage is near the clipping voltage, in case the gate is connected to the supply-voltage V_{dd}. In order to establish the desired bias-current in saturated operation, transistor M6′ has to be resized.

- For correct operation of the current-conveyor, a minimum bias current has to flow through transistor Mc2, to maintain the loop-gain within the conveyor (see section 4.2.2).

- If the output-voltage is low, i.e. clipped to the fixed conveyor-voltage, transistor M5 is cut-off and no current will be able to flow through this transistor. Due to the necessary bias-current for the conveyor, the operation of transistor M6′ might be reversed if current would be drained from the output of the op-amp, resulting in an output-voltage beneath the clipping-voltage. However, the only elements connected to the op-amp are the multipliers of the neighbourhood-cells. In case the input-voltage of the multipliers is equal to their drain-voltage (the output of the op-amp is clipped to the conveyor-voltage), no currents will be drained from the output of the op-amp.

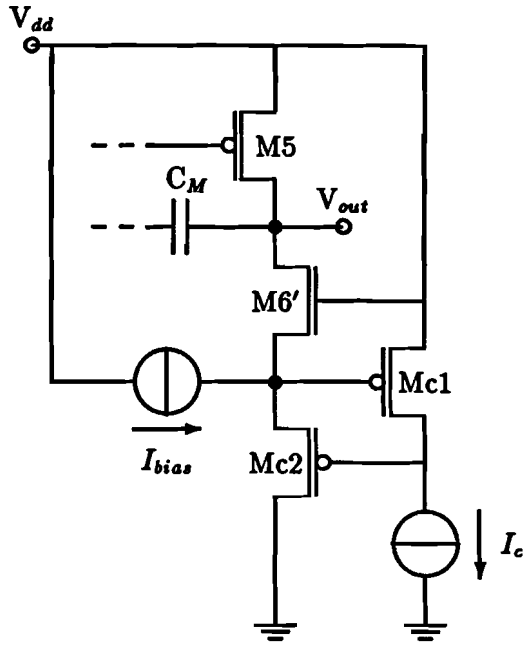

Figure 5.6: A modified op-amp output-stage, creating a minimum output-voltage.

5.3.4 Conclusions

In the design of the standard op-amp many requirements, e.g. common-mode input-range, output-range and slew-rate were given by the application, most of which could be easily met. The requirement for the ratio between slew-rate and gain-bandwidth however brings along some serious problems. In the design of the op-amp an inherent trade-off between increase of the gain-bandwidth and an increase of the slew-rate is present. Trying to achieve both a high gain-bandwidth and high slew-rate can easily result in stability-problems or a considerable decrease in common-mode input-range. As a result, the requirement for the slew-rate could not be met. However, this slew-rate required is a worst-case situation, rarely present in a cell, with all connected multipliers having maximum output. Furthermore, this slew-rate does not change the operation of the cell, but is merely slowing down the cell slightly. Usually, this decrease in response does not change the total operation of the net.

For correct Full Range operation, the output of the op-amp should be clipped to the specified operating range, as both state and output of the cell depend on the output-voltage. Although several 'common' methods are presented no complete solution for this problem has been found. A partial solution, clipping the output-voltage to a lower boundary (determined by a current-conveyor), has however been presented.

5.4 Resistor Design

Apart from the op-amp, the cell-core consists of two multipliers and two resistors. As the multipliers are already discussed in the previous chapter, only the resistors are left to be designed. Although a resistor could easily be made by a linear transistor or even implemented using the bulk-resistance, the design is complicated by strict requirements, e.g. high linearity and small matching errors.

5.4.1 Requirements

The first, and most important requirement concerning the implementation of the resistors is the matching between the two resistors in a single cell. Due to high bias-current which can be present in the input-currents I_L and I_R, a small deviation in resistor-value leads to an offset in the output of the cell-core, which is proportional to this bias-current. The accuracy in matching required can be calculated using the following example.

The currents flowing through resistors R_L and R_R are I_L and I_R respectively. The values of the resistors and currents are

$$\begin{aligned} R_L &= R + \Delta R & \qquad I_L &= I_b + \Delta I \\ R_R &= R & \qquad I_R &= I_b \end{aligned}$$

in which ΔI represents the signal current which has to be detected. The feedback-loop, consisting of the op-amp and multipliers, will add a current ΔI_{fb} to I_R to cancel the difference in *input-voltages* V_L and V_R of the op-amp. Assuming ideal operation of both op-amp and multipliers of the feedback-loop, the voltages V_L and V_R across the resistors in steady state will be identical, and the amount of feedback-current can be calculated:

$$\Delta I_{fb} = \Delta I \left(1 + \frac{\Delta R}{R}\right) + I_b \frac{\Delta R}{R} \tag{5.19}$$

Apart from the scaling of the input-current by a ratio $1+\Delta R/R$, an offset depending on the bias-current I_b (of 19 multipliers) is added to the output ΔI_{fb} of the feedback-loop. Due to the fact that this bias-current can be considerably higher than the maximum difference-current ΔI_{max} (of a single multiplier) which has to be detected, a large error can occur. In order to be able to detect this difference-current, the maximum offset due to the mismatch should be *at least* smaller than ΔI_{max}. As a result, the mismatch between the resistors should not exceed

$$\frac{\Delta R}{R} < \frac{\Delta I_{max}}{I_{b,max}} \approx 1.5\% \tag{5.20}$$

and should preferable be even a small fraction of this value. This, of course, is a very strict requirement which simply can't be satisfied by using plain resistors or transistors.

The second requirement involving the implementation is the linearity of of the resistors. As the time-constant of the cell depends on the value of the resistor (see (5.4)), and the current through the resistors might vary considerably due to the bias-currents, good linearity of the resistors is required.

5.4.2 Design

In order to obtain such a high matching accuracy, special techniques have to be used, as plain bulk-resistors and transistors have considerable variance in parameters even if special layout-techniques are used. An example of a special technique is the so-called *dynamic element switching*. When using this method to match, for example, two transistors, the two elements are inter-changed at a high frequency using switches. For low frequencies, the effective element for both connections equals the average of the two continuously swapped elements. As the switches have to be operated at a much higher frequency than the bandwidth of the system, this technique is not very applicable for CNNs. Another drawback of this technique is the effect of clock-feedthrough on the signals *during* operation. Therefore another technique will be used to match the two resistors.

When using a transistor in linear mode, non-linearities are small if the drain-source voltage is small compared to gate-source voltage:

$$I_d = \frac{\beta}{2}(V_{gs} - V_T - \frac{1}{2}V_{ds}) \cdot V_{ds} \approx \frac{\beta}{2}(V_{gs} - V_T) \cdot V_{ds} \quad \text{if} \quad V_{ds} \ll 2(V_{gs} - V_T) \tag{5.21}$$

The resistance dV_{ds}/dI_d between drain and source of the transistor now depends on the transconductance β, the threshold-voltage V_T, and the gate-source voltage V_{gs}. In order to match two transistors the gate-source voltage can now be used to cancel all deviations in device-parameters. To do so, the difference in resistance has to be measured, and the correct gate-source voltages have to be stored separately for each cell. This *dynamic element matching* can be done using an op-amp, which is already present in the cell-core, and two switching transistors, as is shown in figure 5.7.

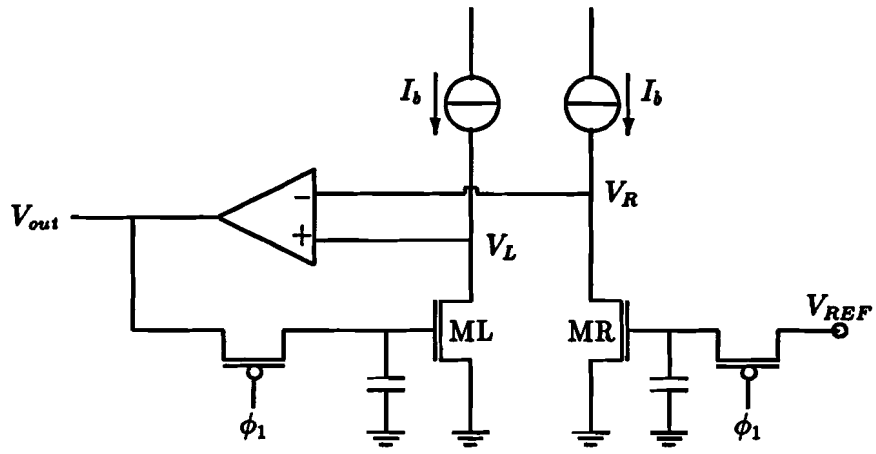

Figure 5.7: Dynamic matching of two resistors.

To match transistors ML and MR, equal currents are sourced into the drains of both transistors. Closing the two switches (ϕ_1 is low) establishes a feedback-loop which adjusts the gate-voltage of ML to cancel the difference between V_L and V_R, thus matching the drain-source resistances of transistors ML and MR. The necessary gate-voltages are stored on small capacitors and the switches can be opened. The main errors when using this method are caused by

- The offset-voltage of the op-amp used. This causes a difference in resistance, proportional to the inverse of the bias-current I_b used. Therefore, a large bias-current should be used.

- The charge-injection on the storage capacitors by the switching transistors. When turning off the switch-transistors, charge stored in on the gate-source and gate-drain capacitance is injected on the storage capacitors, thus increasing the gate-voltages and reducing the resistances. By choosing minimum sized switch-transistors, the charge-injection can be minimized. Moreover, as the gate-drain and gate-source voltages for both switch-transistors will be almost identical, the charge-injection on both storage capacitors will be approximately the same.

- Signal-feedthrough on the storage capacitors. As transistors ML and MR operate in linear mode, the gate-drain capacitance is equal to half the total gate capacitance. This capacitance results in a signal-feedthrough from the drain-voltages V_L and V_R, depending

on the ratio between the storage- and gate-drain capacitance. By choosing small transistors ML and MR, this signal-feedthrough can be minimized. Moreover, signal-feedthrough will be approximately identical for both transistors in case the drain-voltages are close together. Due to the feedback present while the total circuit is operative, these voltages will be identical if the output of the cell-core is *not* saturated. In saturated operation, this signal-feedthrough does not influence the total operation of the cell.

- Current-leakage. As the drain- (or source-) bulk diode in the switch-transistors are reverse biased, a constant leakage-current will flow through the storage-capacitors, increasing the gate voltages. As both switch-transistors will have approximately the same current-leakage, only a small variation in gate-voltages will start to arise. Moreover, the relaxation of the CNN will have ended before any significant change in matching can occur.

5.4.3 Sizing

Using the configuration for the resistors as presented in the previous section, the resistance-transistors ML and MR, and the switch-transistors can be sized. For the latter transistors minimum sizes should be taken to reduce clock-feedthrough, as the voltage-drop across the drain and source of the transistor will very small due to the low gate-voltage, regardless the transconductance of the transistor. To reduce signal-feedthrough, the gate-drain capacitance of transistors ML and MR should be decreased. As the transistors operate in linear mode, this capacitance depends on the total gate-area. Hence, these transistors should also be small. The maximum allowed ratio between width and length can be calculated by substituting the gate-voltage $V_{REF} \approx 4V$, the maximum current $I_{L,R} \approx 120\mu A$ and the maximum allowed drain-voltage $V_{L,R} = 2.5V$ in the transistor equation. The final transistor-sizings are shown in table 5.2.

Table 5.2: Final transistor-sizes for the resistor.

transistor	Width (μm)	Length (μm)
ML	4.8	6.4
MR	4.8	6.4
Mswitch	2.4	2.4

5.4.4 Conclusions

The resistors present in the circuit for the cell-core need to be matched with high accuracy. Usual implementations using special layout-techniques are not sufficient to obtain the required matching. Therefore, a dynamic matching technique is needed to equal the drain-source resistances of two transistors in linear mode. Using a symmetrical design, many errors, e.g. clock- and signal-feedthrough are reduced considerably. For the implementation of this matching technique, little extra circuitry is needed. The op-amp used for matching is already present and correctly connected to the linear transistors. The current sources needed, can be created by using the parallel connected multipliers with *all* gates connected to $V_{w,ref}$, resulting in template-elements equal to zero, and the inputs connected to V_{REF}. Also, no extra reference voltage is needed, as the multiplier reference-voltage $V_{REF} = 4V$ is sufficient to keep the transistors in linear mode (because drain-voltages will not exceed 2.5V). As a result, only *two* switches are

needed to perform the dynamic matching. And, as will be explained in the next chapter, no extra initialization-time for the network is needed.

5.5 Simulations

In order to verify the basic operation, two simulations of the cell-core will be carried out. The DC characteristics will be verified using a single multiplier connected to the differential current-input of the cell-core. By changing the gate-voltage V_z of the feedback-multiplier different slopes of the transfer-function can be achieved as can be concluded from (5.3). To verify the basic dynamic property, i.e. time-constant, of the cell, a transient analysis will be carried out, by again connecting a multiplier to the differential input of the cell-core to analyse the step-response of the cell.

As no complete circuitry was designed to implement the desired clipping function, an ideal circuit element (voltage controlled voltage source) with unity gain and limited output-range was used. Finally, a value for the capacitors present in the cell-core has to be chosen, as this value depends on requirements for the initial-state storage which will be treated in the next chapter. As differential storage is used, these requirements will be small, and small capacitors can be chosen. As a result, the gain-bandwidth of the op-amp will determine the time-constant of the cell, and the effect of the capacitors on the dynamic behaviour will be small. For the simulations in this section a value of $C = 0.1\ pF$ will be chosen. The results of the simulations are shown in figure 5.8 (DC characteristics) and figure 5.9 (step-response).

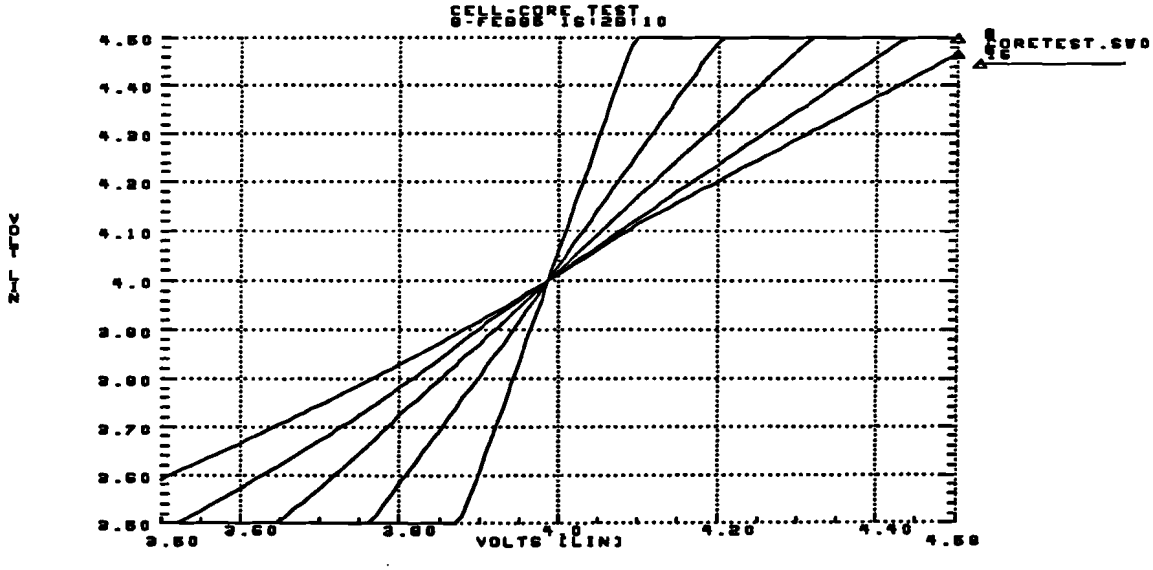

Figure 5.8: DC Transfer-Characteristics of the cell-core for different voltages V_z.

Due to the non-linearity of the feedback-multiplier, as explained in section 4.3 the DC transfer-characteristic also exhibits non-linear behaviour. As a result, the slope of the non-linearity will be lower than expected. Another important inaccuracy of the implemention is the constant offset voltage present in the input of the cell-core. As this offset will be the result of a non-symmetrical configuration in the cell-core circuit, it is most likely to be caused by the output-stage of the op-amp (including the current-mirror M4). Results of experiments to examine this offset have confirmed this assumption.

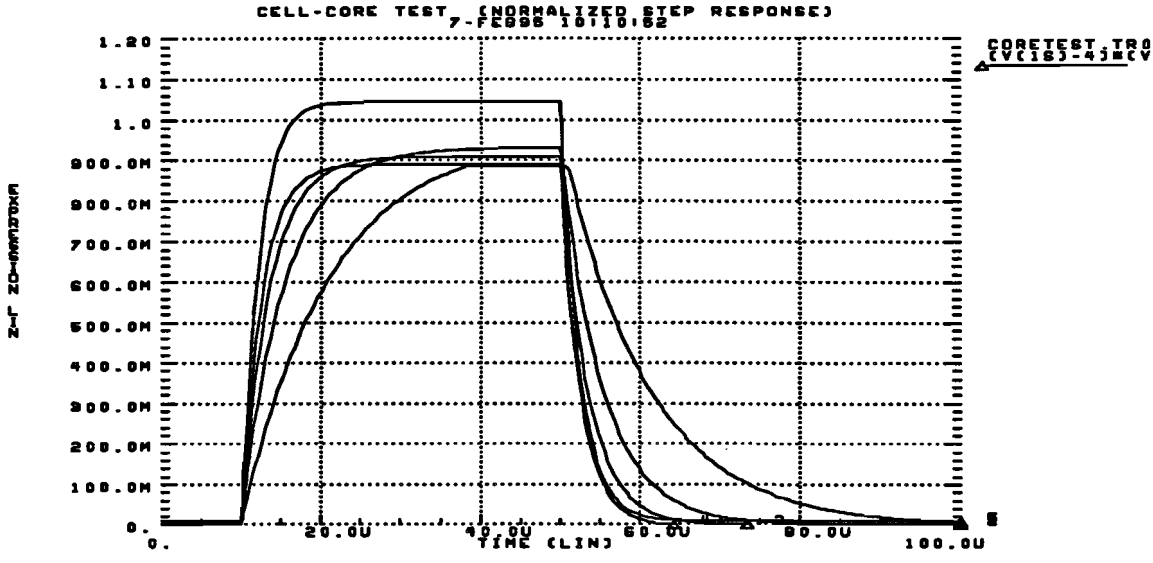

Figure 5.9: Step-response of the cell-core for different voltages V_z.

To be able to verify the results of the step-response-simulation as shown in figure 5.9 the expected time-constants should be determined. The cell time-constant is determined by

$$\tau_{cell} = \frac{C + \tau_{OA}/AR}{\beta_m(V_z - V_{w,ref})}$$

as was derived in section 5.2. By substiting the values for the different parameters, the cell time-constant can be determined for different values of the gate-voltage V_z of the feedback-multiplier. Together with results of the simulation, these values are shown in table 5.3.

Table 5.3: Calculated and simulated cell time-constants.

V_z (V)	theoretical (μs)	simulated (μs)
1.2	14.7	9.9
1.4	7.4	4.6
1.6	4.9	3.4
1.8	3.7	2.6
2.0	3.0	2.1

5.6 Conclusions

In this chapter a cell-core has been presented for the implementation of a two-quadrant full-range CNN. The cell-core consists of an op-amp controlling a feedback-multiplier to cancel the difference in input-currents, which are led through two resistors (connected to the inputs of the op-amp). The dynamics or memory-function of the cell can be implemented by two capacitors or by making use of the gain-bandwidth of the op-amp. To reduce implementation-area, and to obtain a minimum time-constant of the cell, the capacitors could be left out. However, these capacitors will be required to store the initial state as will be shown in the next chapter. The

full-range properties are implemented by limiting the output-range of the op-amp, thus limiting both state and output of the cell, represented by

$$\begin{array}{lll} \text{State:} & V_x & = V_{out} - V_{REF} \\ \text{Output:} & V_y & = V_{out} - V_d \end{array}$$

Although some methods to limit the output-range of the op-amp are presented, a complete solution for clipping the output-voltage was not found. A partial solution for limiting the output-voltage to a lower bound, based on the voltage-stability properties of the current-conveyor (as presented in the previous chapter) and the specific configuration of the op-amp, is however presented. In order to perform simulations and verify the operation of the total cell-core, an ideal circuit element (voltage controlled voltage source) with unity gain and limited output-range is inserted between the output of the op-amp and the output of the cell.

As the input-currents fed through the resistors can contain large bias-currents added to the actual differential signal-current, severe matching-requirements hold for the implementation of the resistors. As layout-techniques for matching will not be sufficient to obtain the desired accuracy, a special *dynamic matching* technique will be used. During an initialization, identical currents are sourced into two transistors in linear mode. Using the op-amp already present in each cell and two minimum sized transistors acting as switches, the impedances of both transistors are matched by adjusting one of the gate-voltages. Due to a symmetrical design, many errors, e.g. clock-feedthrough and signal-feedthrough, will be greatly reduced.

One of the major advantages of this implementation is the direct relation between (a) input-voltages of the multipliers connected and (b) the output-voltage, resulting from the feedback-configuration using the same multipliers. The template-elements will now simply be represented by the ratio of two voltages, and errors will, as opposed to the implementation of Chua and Yang presented in the beginning of this chapter, merely depend on matching between two *identical* elements. A template-element is represented by

$$TE_i = \frac{V_{w,i} - V_{w,ref}}{V_z - V_{w,ref}}$$

A minor drawback resulting from this direct relation is the dependence of the time-constant of the cell on the template scaling-voltage $V_z - V_{w,ref}$

$$\tau_{cell} = \frac{C + \tau_{OA}/AR}{\beta_m(V_z - V_{w,ref})}$$

As this voltage is identical to each cell in the net, it does not influence the operation of the net, but merely changes the operating-speed. In order to make full use of the dynamic-range of the multipliers, $V_z - V_{w,ref}$ should be chosen as large as possible, and is therefore determined by the maximum template-element in a template-set. The operating speed therefore depends on the template-set. Although the cell-core contains an op-amp with clipping-circuitry, two multipliers and four capacitors, the total implementation area will approximately equal the implementation area of the 19 multipliers as

- the capacitors merely function as storage-elements, which can be small due to differential storage and the high operating speed of the net, and
- the number of transistors used will be considerably less than the total number of multiplier-(including conveyor-) transistors (42), which compensates for the fact that some transistors will be larger.

Chapter 6

Cell and Network Control

In this chapter additional circuitry will be presented, required for correct operation of the network and to control the data-flow to and from the network. In order to load the initial state and to apply the input to every individual cell, all inputs u_i should be available. Similarly, all outputs y_i should be available to read out the results of the relaxation of the network. In case of small CNNs (typ. up to 40 cells), this will not result in major problems concerning the packaging of the CNN-chip. However, in large CNNs some sort of cell-addressing, e.g. column- or row-addressing needs to be used for the data-transfer to and from the cells. This causes need for three additional circuits: (1) circuits for the actual cell-addressing, (2) circuits to locally store the input-value for each cell and (3) a circuit to break the feedback-loop in each cell to retain the initial state while applying the input u_i to each cell, as mentioned in section 3.1.

6.1 Operation Procedure

In order to operate a complete CNN four main tasks have to be performed. First, the network has to be initialized, i.e. an initial state $x_i(0)$ has to be applied to each cell, which can be done using an initalizing template-set as described in section 3.1. This template-set might be applied externally, but can also be created using switches, as only template-elements of one and zero will be used. These values can be created by connecting the template-scaling input to $V_z = 2.0V$ (ensuring maximum template values equal to one), connecting a weight-voltage to the reference-voltage $V_w = V_{w,ref}$ to create a 'zero' or connecting it to ground to create a 'one'. As the template-set is space-invariant, only 2×19 switches are needed to apply the initialization template-set internally (19), and to apply the operation template-set afterwards (19).

Secondly, the input u_i for each cell has to be applied to the local storage circuit, while retaining the initial state in each cell. This storage circuit needs to be able to retain the stored value during the the total period of initialization, relaxation and output read-out of the total network, without significant change (loss). More-over, as this storage-circuit is connected to nine multipliers (of the total neighbourhood), it should be able to supply considerable current, again without significant change in the stored value.

The third main task consists of simply applying the correct operation template-set, and starting the relaxation of the network by restoring the broken feedback in each cell.

Finally, the output of all cells needs to be read. As the local positive feedback (created by the correct template-set) retains the output after the network has reached the final state (the relaxation has ended), no extra storage circuits are needed. However, the read-out time for the total network is bounded by change of the input-values on the local storage circuits. A large change in these input-values might cause a cell to change it's state again, restarting the total relaxation process, resulting in erroneous behaviour of the network. Using these main tasks the total operation procedure can now determined:

1. The initial state of each cell has to be applied to the local input-storage circuit. During this period the two resistors (implemented by linear transistors) have to be matched. As equal currents need to be fed to the two transistors, the weights of all multipliers should be set to zero, requiring two extra switches for the weights $B_{i,i}$ and I, as other weights already have the necessary switches to create zero weight for the initalization template-set. Although not necessary, the input of all cells could be connected to a constant voltage,

e.g. V_{REF} to create a sufficiently large current through the resistors, requiring two extra switches per cell (one to connect to the storage circuit, and one to connect to the constant voltage). As these currents through the resistors will not depend on the applied voltage on the storage-circuit, the matching of the resistors will be improved.

2. The initialization template-set has to be applied to the cells (either externally or internally), the inputs of the multipliers should be connected to the storage-circuit, and the relaxation of the cells to apply the initial state can be started.

3. To retain the initial state in each cell, the feedback-loop should be broken, which will be discussed in the next section. Now the input-values u_i have to be applied to the local storage-circuit in each cell.

4. As the network has now been completely set-up for operation, the correct operation template-set can be applied and the relaxation can be started by restoring the feedback-loop in each cell.

5. After the network has reached it's final state[1], the outputs of the cells can be read.

6.2 Feedback control

To retain the initial state during the following set-up procedures as described in the previous section, the feedback loop should be broken to avoid an exponential decay of the initial state (as result of the RC pair). As the state is represented by the output of an op-amp $V_{out} - V_{REF}$, capacitors should be used to store this initial state, as shown in figure 5.2. Due to the limitations on the magnitude of the initial state $|x(0)|$ (see section 2.4) the output of the op-amp is not saturated and the inputs V_L and V_R will be identical. Hence the state will be stored on the two capacitors, retaining a voltage $V_{out} - V_R$ and V_L respectively. In order to retain these voltages, no current should be able to flow through both capacitors. As a result, the inputs of the op-amp and the two resistors should be separated by switches, as shown in figure 6.1.

Errors in the true initial state, i.e. the state-value at the exact moment the template-set becomes operational, are similar to the ones when using the switches for matching the two resistors (see section 5.4.2):

- current-leakage through the reverse-biased bulk-source diodes.
- Clock-feedthrough due to charge-injection.

Because of the symmetrical design these errors will be small and will have no effect on the total behaviour of the cell. However, to determine the exact influence on the initial state, simulations and measurements should be carried out.

6.3 Input Storage

In order to apply an input to each cell, this input has to be stored locally in case a large network is used and not all inputs are available at package-pins directly. As the inputs are connected to the multipliers of all (nine) cells of the neighbourhood, some extra circuitry is needed to to supply the input-currents when storing a value on a capacitor. As these currents are large, the output-impedance should be low to reduce the resulting voltage-drop. This low impedance

[1] the total relaxation time depends on the used template-set (template-scaling input and the signal propagating properties of the specific operation).

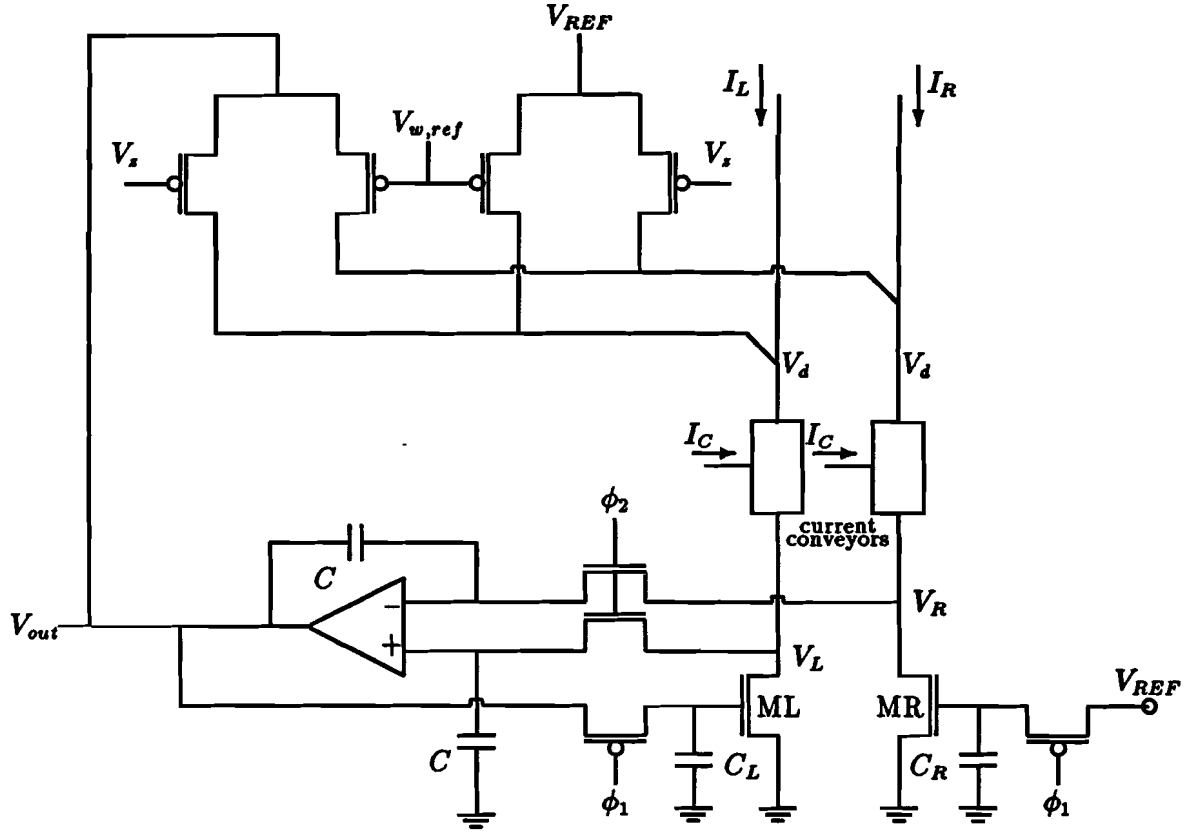

Figure 6.1: The complete cell-core with control-switches.

can be achieved by using a negative feedback to compensate for this voltage drop. A simple solution to this problem, including all control-switches is shown in figure 6.2. In this circuit the input-voltage is stored on C_{in} and a differential stage with unity-gain feedback is used to supply the necessary input-current without a significant voltage-drop.

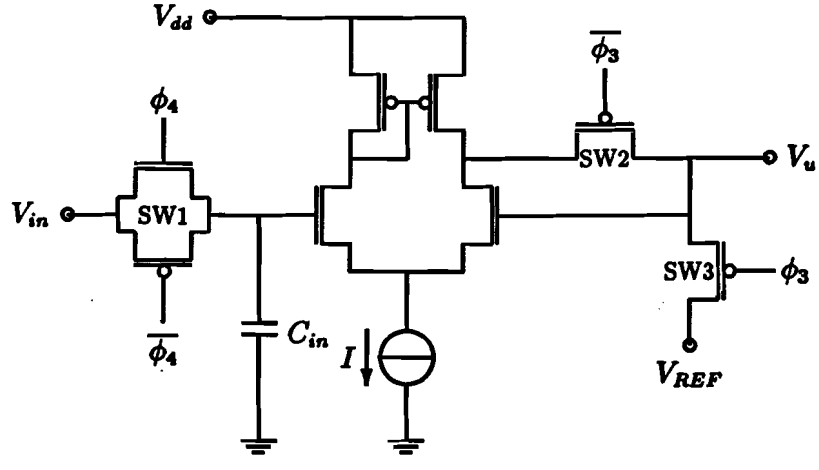

Figure 6.2: Storage circuit (with control-switches) having low output-impedance.

Switch SW1 is controlled by the cell-addressing circuitry and is used to store the input-value. Switch SW2 creates the desired feedback, and connects the storage-circuit with the multipliers for operation. Finally switch SW3 is used to connect the input-multipliers to a fixed reference voltage V_{REF} to create the desired currents for the dynamic resistor matching. When designing this circuit, several aspects have to be taken into account:

- The input- and output-voltage range equals $3.5V \leq V_{in,u} \leq 4.5V$.
- The maximum required output-current equals $I_u \leq 100\mu A$.
- In order to reduce the effects of current-leakage through capacitor C_{in}, this capacitor should be reasonably large, and the transistors of switch SW1 should be as small as possible.
- In order to reduce the effects of clock-feedthrough on capacitor C_{in}, a well designed CMOS switch or a dummy transistor should be used for switch SW1 to minimize the charge-injection.
- Although the transistor of switch SW3 operate in linear mode, they should be large (wide) enough to allow the necessary input-currents without creating a significant voltage-drop.

6.4 Cell Addressing

Because the number of pins on a chip-package e.g. DIL (dual-in-line) or PGA (pin-grid-array) is limited, not all inputs and outputs of a reasonably (over 40 cells) sized CNN can be connected directly, some sort of cell-addressing needs to be used. Although individual addressing of each cell is possible, this unnecessarily hampers the design and increases the total processing time. As the cells are organized on a (square) grid, column- or row-addressing might be used. In this way, the number of pins required can be reduced considerably. As the address-decoding has to be done only once per column or row, the circuits required result in only a marginal overhead in implementation area of the complete CNN. The total number of pins (NOP) of a *square* CNN with R rows and columns can be calculated. Besides the pins required for the supply-voltage (2) and the template-set (19) extra pins are required for reference-voltages (2) and the template-scaling input (1). For each row the input and output of a cell has to be connected, resulting in $2R$ pins. Finally, $\lceil {}^2\log R \rceil$ pins are required to address all columns[2] and three pins (ϕ_1, ϕ_2 and ϕ_3) are necessary to control the data-flow in the network. This results in the total number of pins required:

$$NOP = 2R + \lceil {}^2\log R \rceil + 27$$

6.5 Conclusions

In this chapter additional circuitry has been presented for cell and network control. Most circuitry is needed due to the limited number of pins available on a chip-package. Apart from the circuit for local storage of the input, in which a feedback-loop is used to reduce the voltage-drop (resulting from the input-current of nine multipliers), seven[3] switches are needed in a cell to completely control a cell. Finally, 21 switches are needed to create the required initialization template-set internally. Although the basic circuits have been presented, the actual design and simulations still have to be carried out to determine accuracy and chip area required.

[2] $\lceil {}^2\log R \rceil$ means the nearest integer number higher than ${}^2\log R$. As a result, all columns can be addressed using this number of pins for a binary address.

[3] Dynamic resistor matching: three switches, feedback-control: two switches and input-control/cell-selection: two switches.

Chapter 7

Total Cell Circuit

In this chapter, the total cell circuit will be described and it's main features, e.g. area, power-consumption and speed highlighted. Also a simulation will be presented of the Connected Component Detection template-set.

7.1 Description

The total cell circuit can be constructed by combining the multiplier-section (figure 4.7), the cell-core (figure 6.1) and the input-storage circuit (figure 6.2). The switches to control the template-elements are deliberately left out, as they need to be implemented only once in a complete CNN and aren't therefore part of any cell. The total cell circuit is shown in appendix D. The basic elements of the cell, e.g. multiplier, current-conveyor, storage-circuit and the cell-core are clearly recognizable. For this total cell circuit, some main features can be determined, which are shown in table 7.1.[1] [2] [3]

Table 7.1: Main features of the total cell circuit.

feature	value	unit
# of transistors[1]	80	-
active area (approx.)[2]	$45 \cdot 10^3$	μm^2
# cells/mm^2 (est.)[3]	10	-
power consumption (max.)	1.5	mW
time-constant (min.)	2.1	μs

7.2 Template Transformation

Based on the cell-core structure and the defined input- and output ranges, the template-sets to be used for this 2-quadrant CNN can be calculated (or transformed) from the original template-sets for 4-quadrant CNNs. In section 5.2 the input- and output ranges for this CNN were derived:

$$\begin{aligned} -\tfrac{1}{2} \;&\leq\; x_i \;\leq\; \tfrac{1}{2} \\ 0 \;&\leq\; y_i \;\leq\; 1 \end{aligned}$$

Substituting these boundaries into (2.18) the new template-set to be used to achieve the same operation can be calculated:

$$\begin{aligned} A_{\text{new}} &= A_{\text{old}} \\ B_{\text{new}} &= B_{\text{old}} \\ I_{\text{new}} &= \frac{1}{2}(I_{\text{old}} - \sum_{k \in N_r(i)} A_{\text{old},i,k} + B_{\text{old},i,k}) \end{aligned} \tag{7.1}$$

[1] Total number of transistors in a cell, when an op-amp *without* clipping circuitry is used.

[2] This approximation is based on area needed for each transistor (including transistor-spacing and two contact-holes): $(L + 19.4)(W + 5)$ μm^2 and does not include wiring.

[3] This estimation is based on the assumption that the wiring needed in a cell will occupy approximately the same area as the transistors in a cell.

7.3 Example: CCD

In order to verify the total operation of the designed cell, a small network will be simulated. As the Connected Component Detection template-set, as introduced in section 2.4.2, operates in a single direction, allowing a small network, nine cells connected in line will be used. Moreover, this template-set allows an excellent examination of the propagation-effects in a CNN. The setup of this simulation is shown in figure 7.1.

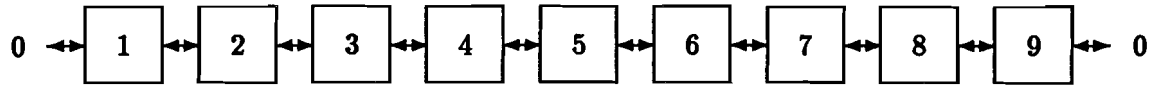

Figure 7.1: A small CNN for CCD simulation.

Apart from the connections of the inner cells with left and right neighbours, an important aspect of this set-up is the connection to a bias-value for the two edge-cells. As opposed to other templates the multipliers belonging to the non-existing neighbours *are* present and connected to a bias-value. For the specific operation of CCD, the remaining inputs of the edge-cells should be connected to the *minimum* output-value $y_{min} = 0$ (or 3.5V). The transient simulation (hspice file in appendix E) shown in figure 7.2 now consists of three steps:

1. Matching the two resistors (0-10μs).
2. Initializing the cells using the initialization-template (10-20μs).
3. Relaxation of the network (20-60μs).

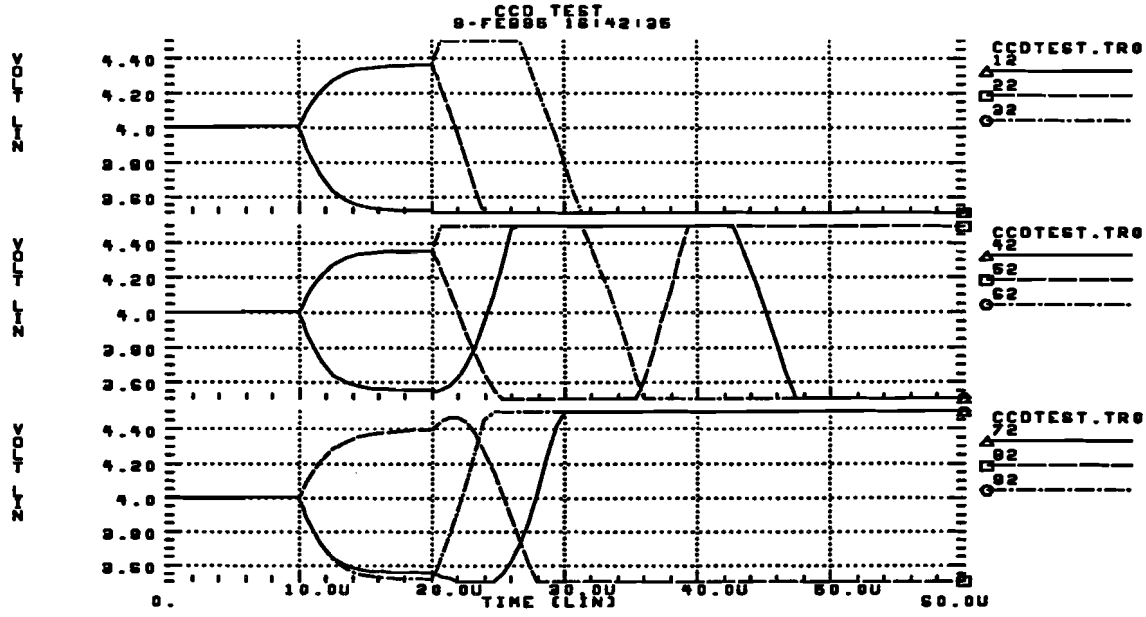

Figure 7.2: Transient simulation of a nine-cell CNN with CCD-template.

By examining the outputs of the cells at different time-instances and representing them by binary signs as introduced in section 2.4.2 the operation of the CCD can be clarified, as can be seen in figure 7.3.

time	Cell								
(μs)	1	2	3	4	5	6	7	8	9
20.0	·	●	●	·	●	●	·	●	·
23.5	·	·	●	·	·	●	·	●	●
27.0	·	·	●	●	·	●	·	·	●
30.5	·	·	·	●	·	●	●	·	●
35.0	·	·	·	●	·	·	●	·	●
39.5	·	·	·	●	●	·	●	·	●
47.0	·	·	·	·	●	·	●	·	●

Figure 7.3: Binary representation of the outputs of the CCD-CNN.

Apart from the obvious propagation-effects creating the global processing-capabilities, some other properties can be derived from this simulation. As the maximum template-value equals two, the time-constant of the cells should be approximately 2×2.1 μs, as the template-scaling input changes this time-constant. This value is verified by figure 7.2. Another property already mentioned with simulations of the cell-core in section 5.5, is the error in the cell-core due to non-linearity of the feedback-multipliers. This error causes the initial-state values at the end of the initialization-period (t=20 μs) to deviate up to 30% from the actual input-values (3.5V and 4.5V). In case of the CCD-template-set this does not influence the operation, but might create problems when using template-sets more sensitive to distortions.

Chapter 8

Conclusions and Recommendations

Conclusions

In this report a design is presented for an analogue programmable Cellular Neural Network (CNN). Based on the original idea of Chua and Yang [1] a Full-Range implementation (see [2]) of a Linear-Cloning-Template CNN on a square grid has been made. An important new feature of this design is the use of 2-quadrant multipliers and positive valued inputs and outputs for the implementation of the variable template-elements. As a result, the total implementation area can be reduced significantly.

For the implementation of the 2-quadrant multiplication, the so-called two-transistor multiplier is used, consisting of two transistors in linear mode, having a common source and identical drain-voltages. The output of the multiplier consists of the difference between the two drain-currents. However, the two-transistor multiplier has a major drawback. As the outputs of several multipliers have to be added in a CNN-cell, the differential outputs of the multipliers can be connected together. However, due to a large bias present in each output of a multiplier, the *difference* between the two summed currents can be considerably smaller than the total bias. This results in poor accuracy and severe requirements on other circuit-elements.

Another important aspect of the design is the implementation of the degrading memory (lossy integrator) and non-linearity of the cell. This so-called cell-core is implemented using an op-amp with limited output-range and a multiplier in a feedback-configuration. Using two resistors, the differential input-current is measured and a compensating differential current is added using the feedback-multiplier. As the differential input current equals the sum of outputs of identical multipliers, a direct relation between an input-voltage and the output-voltage of a cell (identical to a template-element) is created. A template-element now solely depends on the ratio between two voltages, and is therefore easily determined.

The cell time-constant resulting from this cell-core depends on the unity-gain bandwidth of the op-amp, additional capacitors (only needed for storage of the initial state) and the feedback-loop gain. Although the gain-bandwidth of an op-amp can be high, strict requirements for the slew-rate of the op-amp limit this frequency. However the main limitation of the cell time-constant is the result of the feedback-loop gain. This gain, consisting of the trans-conductance of the feedback-multiplier and the resistors, is small due to the limited value of the latter elements. This is the result of the large bias currents present in the output of the parallel connected multipliers (19 in each cell). Another aspect of this gain is the dependence on the trans-conductance of the feedback-multiplier. As this trans-conductance is also used to scale the template-elements, the cell time-constant depends on the template-set used. This however does not influence the operation of the net but merely changes the speed at which it operates, as the scaling-input will be identical for each cell in the net.

Although most of the cell-circuitry has been designed, some problems still need to be solved. The main problem consists of creating a limited output-range for the op-amp. Although some possibilities have been presented, a complete solution has not been found. Another problem is the design of the cell-control circuitry. Although the basic circuits have been designed, the exact sizing of elements and necessary simulations still have to be done.

Recommendations

Based on findings, resulting from designing a Cellular Neural Network, the following recommendations can be made:

1. Although a reduction in implementation area can be achieved by adding currents of the multipliers differentially, it can result poor accuracy and limitations for other elements due to the bias present in these currents. Therefore these bias-currents should be reduced, or a multiplier with single ended output should be used (by substracting differential currents individually). The latter solution will require more transistors, but will also reduce the number of intercell-connections.

2. As no exact accuracy-requirements are known for the implementation of a CNN, an investigation to accuracy-analysis and/or learning capabilities of CNNs might be considered.

3. The input-current needed for the two-transistor multipliers constitutes a major problem for the implementation of the op-amp. The large output-current needed for the op-amp not only requires large transistors, it also inhibits the implementation of the non-linearity by clipping to the supply-voltage. Therefore, other multipliers might be (re-)considered.

4. In order to examine the basic properties of CNNs, an implementation of a (not necessarily analogue) programmable CNN could be made, with each cell having only four neighbours (North, East, South and West), resulting in a considerably improved accuracy. This could also be achieved in the currently designed CNN by using two weight reference-voltages $V_{w,ref}$, allowing to completely disable (cut off) the diagonal neighbours.

Acknowledgements

First I would like to thank my coach, Hans Hegt, for his all his support, but especially for his way of coaching, leaving me freedom to experiment while helping me taking different perspectives to the problems we met. I'd also like to thank my "assistent-coach", Paul Bruin, for his useful ideas and discussions throughout my graduation project, and all other (AIO-)members of group EEB.

I'd like to thank my fellow-student and friend, Rick, for helping me keeping faith in accomplishing this study and making it more fun(?), and Arno and Rogier for their "*Always look on the bright side of life...*"

Last but not least I'd like to thank my parents and brother for their support throughout my study and for putting up with my stress during "extended working hours."

References

[1] L. Chua and L. Yang, "Cellular neural networks: Theory," *IEEE Transactions on circuits and systems*, vol. CAS-35, no. 10, pp. 1257–1272, 1988.

[2] A. Rodriguez-Vázquez, S. Espejo, R. Dominguez-Castro, J. Huertas, and E. Sánchez-Sinencio, "Current-mode techniques for the implementation of continuous- and discrete-time cellular neural networks," *IEEE Transactions on circuits and systems II*, vol. CAS-40, no. 3, pp. 132–146, 1993.

[3] L. Chua and T. Roska, "The cnn paradigm," *IEEE Transactions on circuits and systems II*, vol. CAS-40, no. 3, pp. 147–156, 1993.

[4] V. Cimagalli and M. Balsi, "Cellular neural networks: A review," in *Proc. of Sixth Italian Workshop on Parallel Architectures and Neural Networks* (E. Caianiello, ed.), World Scientific, 1993.

[5] J. Nossek and G. Seiler, "Cellular neural networks: Theory and circuit design," *Int. Journal of Circuit Theory and Applications*, vol. 20, pp. 533–553, 1992.

[6] L. Jiao, "Fully integrated continuous-time neural networks," in *IEEE Proc. of Int. Symp. on Circuits and Systems*, vol. 3, pp. 2108–2111, Piscataway: IEEE, Inc., 1989.

[7] R. Perfetti, "On the op-amp based circuit design of cellular neural networks," *Int. Journal of Circuit Theory and Applications*, vol. 22, pp. 425–430, 1994.

[8] P. Kinget and M. Steyaert, "Impact of system specifications on analogue cmos implementations of continuously programmable cellular neural networks," in *Int. Conference on Neural Networks ICNN*, vol. III, pp. 1949–1954, 1994.

[9] D. Ribner and M. Copeland, "Design techniques for cascoded cmos op-amps with improved psrr and common-mode input range," *IEEE Journal of Solid State Circuits*, vol. SC-19, pp. 919–925, 1984.

Appendix A

Transformation of the saturation function

In order to compare the cell dynamics of the 2-Quadrant CNN, as described in section 2.4, with the general saturation function to the dynamics of the original cell, a transformation is used to 'restore' the original saturation function. In order to make the equations clearer, the time index (t) is omitted.

$$\frac{dx_i}{dt} = -x_i + \sum_{k \in N_r(i)} A_{i,k} y_k + \sum_{k \in N_r(i)} B_{i,k} u_k + I_i$$

with

$$y_i = \frac{b-a}{4}\left\{\left|\frac{2}{d-c}x_i - \frac{c+d}{d-c} + 1\right| - \left|\frac{2}{d-c}x_i - \frac{c+d}{c-d} - 1\right|\right\} + \frac{a+b}{2}$$

The following transformations are now used to restore the original saturation function:

$$\begin{cases} x_i = \frac{d-c}{2}x'_i + \frac{c+d}{2} \\ y_i = \frac{b-a}{2}y'_i + \frac{a+b}{2} \\ u_i = \frac{b-a}{2}u'_i + \frac{a+b}{2} \end{cases}$$

Note that the relation between y'_i and x'_i is defined by the *original* saturation function Chua and Yang specified. These transformations result in

$$\begin{aligned} \frac{d\left\{\frac{d-c}{2}x'_i + \frac{c+d}{2}\right\}}{dt} = & -\frac{d-c}{2}x'_i - \frac{c+d}{2} \\ & + \sum_{k \in N_r(i)} A_{i,k}\left[\frac{b-a}{4}\{|x'_k+1| - |x'_k-1|\} + \frac{a+b}{2}\right] \\ & + \sum_{k \in N_r(i)} B_{i,k}\left[\frac{b-a}{2}u'_k + \frac{a+b}{2}\right] + I_i \end{aligned}$$

Multiplying both sides with $\frac{2}{d-c}$ gives:

$$\begin{aligned} \frac{d\left\{x'_i + \frac{c+d}{d-c}\right\}}{dt} = & -x'_i - \left(\frac{c+d}{d-c}\right) \\ & + \sum_{k \in N_r(i)}\left(\frac{b-a}{d-c}\right)A_{i,k} \cdot \frac{1}{2}\{|x'_k+1| - |x'_k-1|\} + \sum_{k \in N_r(i)}\left(\frac{b-a}{d-c}\right)A_{i,k} \\ & + \sum_{k \in N_r(i)}\left(\frac{b-a}{d-c}\right)B_{i,k}u'_k + \sum_{k \in N_r(i)}\left(\frac{b-a}{d-c}\right)B_{i,k} + \frac{2}{d-c}I_i \end{aligned}$$

in which the saturation function can be replaced by y'_i:

$$\begin{aligned} \frac{d\left\{x'_i + \frac{c+d}{d-c}\right\}}{dt} = & -x'_i(t) + \sum_{k \in N_r(i)}\left(\frac{b-a}{d-c}\right)A_{i,k}y'_k \\ & + \sum_{k \in N_r(i)}\left(\frac{b-a}{d-c}\right)B_{i,k}u'_k \\ & + \left\{\left(\frac{2}{d-c}\right)I_i - \frac{c+d}{d-c} + \sum_{k \in N_r(i)}\left(\frac{a+b}{d-c}\right)(A_{i,k}+B_{i,k})\right\} \end{aligned}$$

Appendix B

Cmos Level-2 Parameters

Table B.1: Transistor parameters of the CMOS process at IMEC.

parameter	symbol	NMOS	PMOS	unit
Model selector	LEVEL	2	2	
Channel length modulation	LAMBDA*	0.046	0.027	1/V
Oxide thickness	TOX	$42.5 \cdot 10^{-9}$	$42.5 \cdot 10^{-9}$	m
Threshold voltage	VTO	0.85	-0.85	V
Bulk surface doping	NSUB	$1.16 \cdot 10^{15}$	$9.3 \cdot 10^{15}$	cm^{-3}
Carrier mobility	UO	620	210	$cm^2/V{\cdot}s$
Fast surface state density	NFS	$2.64{\cdot}10^{11}$	$1.36{\cdot}10^{11}$	cm−2
Mobility critical field	UCRIT	$5.2{\cdot}10^{4}$	$9.4{\cdot}10^{4}$	V/cm
Critical field exponent	UEXP	0.104	0.286	
Junction depth	XJ	$0.3{\cdot}10^{-6}$	$0.05{\cdot}10^{-6}$	m
Narrow width factor	DELTA	1.43	1.93	
Diffusion sheet resistance	RSH	165	350	Ohm/sq
Saturation current	JS	$1{\cdot}10^{-3}$	$1{\cdot}10^{-3}$	A/m^2
Lateral length diffusion	LD	0	$0.25{\cdot}10^{-6}$	m
Lateral width diffusion	WD	$0.1{\cdot}10^{-6}$	$0.2{\cdot}10^{-6}$	m
length reduction (etching)	DELL	0	0	m
width reduction (etching)	DELW	0	0	m
Gate source overlap capacitance	CGSO	$0.8 \cdot 10^{-10}$	$2.8 \cdot 10^{-10}$	F/m
Gate drain overlap capacitance	CGDO	$0.8 \cdot 10^{-10}$	$2.8 \cdot 10^{-10}$	F/m
Bulk bottom capacitance	CJ	$0.9 \cdot 10^{-4}$	$3.2 \cdot 10^{-4}$	F/m^2
Bulk sidewall capacitance	CJSW	$3.3 \cdot 10^{-10}$	$4.0 \cdot 10^{-4}$	F/m
bottom grading coefficient	MJ	0.5	0.5	
sidewall grading coefficient	MJSW	0.33	0.33	
bottom potential	PB	0.8	0.8	V
capacitance coefficient	FC	0.5	0.5	
flicker noise coefficient	KF	$1 \cdot 10^{-28}$	$3.0 \cdot 10^{-30}$	
flicker noise exponent	AF	1	1	

* The channel length modulation or early-effect depends on the length of the transistor. The values given are those for transistors with a length of 3 μm. To calculate the *lambda* for different transistor-lengths the following formula applies:

$$\text{LAMBDA} = \text{LAMBDA}_{L=3\mu m} \cdot \frac{3u - 2 * LD + DELL}{L - 2 * LD + DELL}$$

For every transistor with a different length, a different model should be specified. However, when using HSPICE, automatic selection is possible using sub-models and model-selection criteria. Sub-models are created by using a single model name "XXX" with different extensions, each starting with a period, e.g. ".sub1" and ".L30mu". The model-selection criteria added to each separate sub-model description are minimum and maximum length (LMIN and LMAX respectively) and/or minimum and maximum width (WMIN and WMAX respectively).

For the design of capacitances and resistances the following parameters are given.

Table B.2: General parameters of the CMOS process at IMEC.

Capacitance		unit
poly-1 to poly-2	0.505	$fF/\mu m^2$
poly-1 to metal-2	0.048	$fF/\mu m^2$
Resistance		**unit**
Poly-1	17.7	Ohm/sq
Poly-2	18.2	Ohm/sq
P-diffusion	37.4	Ohm/sq
N-diffusion	28.7	Ohm/sq
N-well	$2.2 \cdot 10^3$	Ohm/sq

Appendix C

Cell Core Dynamics

To implement the cell-dynamics and the output-function of a CNN-cell, the so-called cell-core has to be designed. In this part of the cell, the total cell-input $\sum A_i \cdot y_i + \sum B_i \cdot u_i + I_i$ has to be fed to a lossy integrator resulting in a state x_i, from which the output y_i is determined using a non-linear function. The cell-core for a Full-Range implementation of a CNN using the 2T-multiplier is shown in figure C.1.

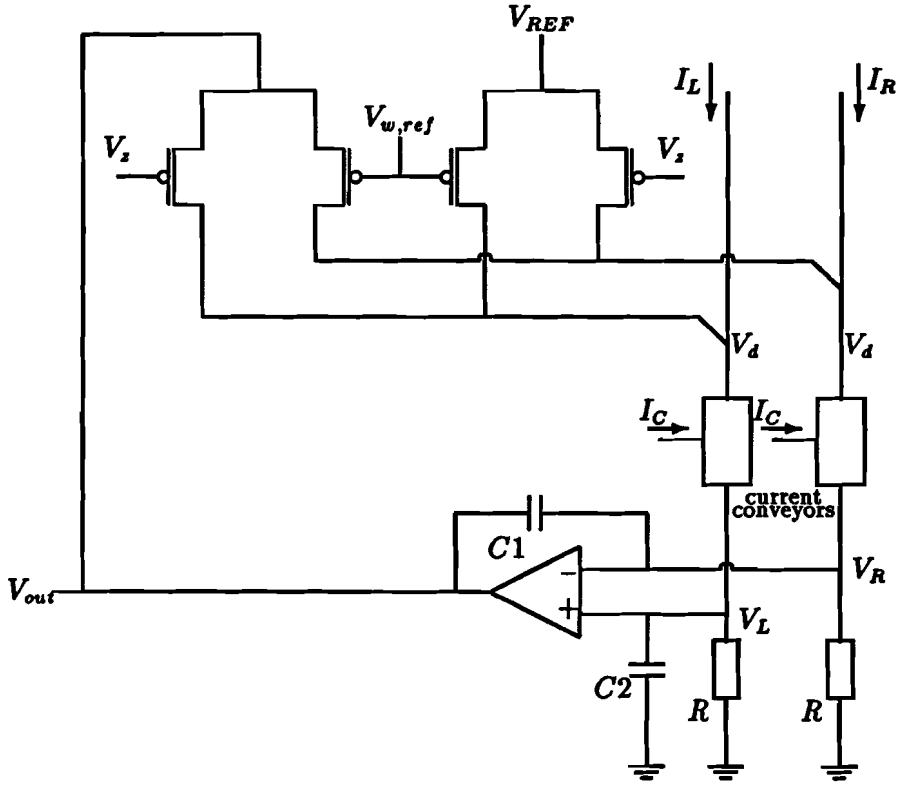

Figure C.1: A cell-core for a Full-Range CNN using 2T-multipliers.

Using a non-ideal op-amp with amplification A and a dominant time-constant τ_{OA} the output V_{out} of the op-amp equals

$$V_{out} = \begin{cases} A \cdot (V_L - V_R) - \tau_{OA} \frac{dV_{out}}{dt} & \text{not clipped} \\ V_{SAT_{LO,HI}} & \text{clipped (High or Low)} \end{cases} \quad \text{(C.1)}$$

The input-voltages of the op-amp are determined by the currents I_L and I_R, the currents of the feedback-multipliers and the currents of the capacitors $C1 = C2 = C$, all flowing through the resistors R. The currents in the capacitors are

$$I_{C1} = C \cdot \frac{d(V_{out} - V_R)}{dt} \quad \text{(C.2)}$$

$$I_{C2} = C \cdot \frac{dV_L}{dt} \quad \text{(C.3)}$$

not clipped:

Substituting all currents into (C.1) for the not-clipped situation results in

$$\begin{aligned} V_{out} &= AR[I_L - I_R - \beta_m(V_{out} - V_d)(V_z - V_{w,ref}) - \beta_m(V_{REF} - V_d)(V_{w,ref} - V_z)] \\ &\quad + AR\left[-C\frac{dV_L}{dt} - C\frac{d(V_{out} - V_R)}{dt}\right] - \tau_{OA}\frac{dV_{out}}{dt} \end{aligned} \tag{C.4}$$

Substituting V_L using (C.1) and working out this equation results in the following *second-order* differential equation:

$$\frac{C\tau_{OA}}{A}\cdot\frac{d^2V_{out}}{dt^2} + \left[\frac{A+1}{A}C + \frac{\tau_{OA}}{AR}\right]\cdot\frac{dV_{out}}{dt} = -\beta_m(V_{out} - V_{REF})(V_z - V_{w,ref}) + I_L - I_R - \frac{V_{out}}{AR} \tag{C.5}$$

Using the general form of a second-order differential equation with two time-constants

$$\tau_1\tau_2\frac{d^2x}{dt^2} + (\tau_1 + \tau_2)\frac{dx}{dt} + 1 = \text{constant}$$

and assuming that $2 \cdot \beta_m \cdot (V_z - V_{w,ref}) \cdot R \ll 1$ (which can be verified afterwards) shows that the dynamics of the cell-core is governed by two time-constants

$$\tau_1 \approx \frac{C + \tau_{OA}/AR}{\beta_m(V_z - V_{w,ref})} \tag{C.6}$$

$$\tau_2 \approx 0 \tag{C.7}$$

As there is a dominating time-constant $\tau_1 \gg \tau_2$ equation (C.5) can be approximated by the *first-order* differential equation

$$\left[C + \frac{\tau_{OA}}{AR}\right]\cdot\frac{dV_{out}}{dt} = -\beta_m(V_{out} - V_{REF})(V_z - V_{w,ref}) + I_L - I_R \tag{C.8}$$

Finally, writing $I_L - I_R$ as the sum of outputs of the multipliers, this equation can be rewritten as

$$\left[\frac{C + \tau_{OA}/AR}{\beta_m(V_z - V_{w,ref})}\right]\cdot\frac{dV_{out}}{dt} = -(V_{out} - V_{REF}) + \sum_i \frac{(V_{w,i} - V_{w,ref})}{(V_z - V_{w,ref})}\cdot(V_{in,i} - V_d) \tag{C.9}$$

clipped:

As the output is now clipped, a change in the input-currents I_L and I_R will not be cancelled anymore, and will lead to a change in the op-amp input-voltage $V_L - V_R$. Again, writing out all currents, the following dynamic equation can be obtained:

$$V_L - V_R = R\left[I_L - I_R - \beta_m(V_{SAT} - V_{REF})(V_z - V_{w,ref}) - C\frac{dV_L}{dt} - C\frac{d(V_{SAT} - V_{REF})}{dt}\right] \tag{C.10}$$

Re-arranging this equation results in

$$C\frac{d(V_L - V_R)}{dt} = -\frac{(V_L - V_R)}{R} - \beta_m(V_{SAT} - V_{REF})(V_z - V_{w,ref}) + I_L - I_R \tag{C.11}$$

Appendix D

Total Cell Circuit

In figure D.1 the total cell circuit is depicted. The different parts, e.g. storage circuit, multipliers, cell-core and current-conveyors are clearly recognizable.

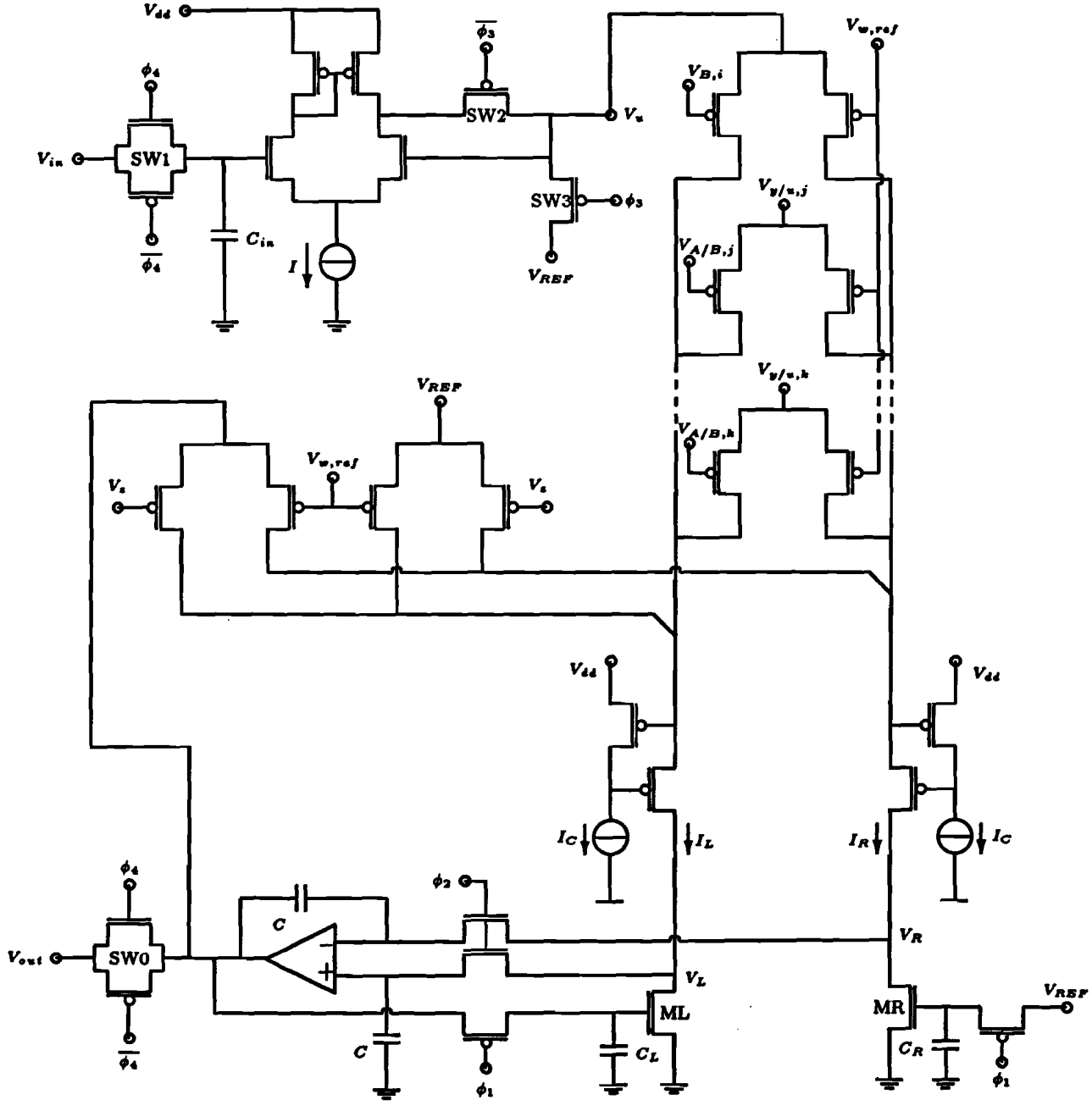

Figure D.1: Total Cell Circuit.

Appendix E

Hspice Listing CCD-test

```
CCD test
.include 'alltyp.mdl'
*
*** OPAMP
*              in+ in- out vdd
.SUBCKT OPAMP  3   2   12  1
*dgsb
M1a 5 2 4 4 P W=4.8U L=5.6U AS=39e-12 AD=39e-12 PS=25.6U PD=25.6U
M1b 6 3 4 4 P W=4.8U L=5.6U AS=39e-12 AD=39e-12 PS=25.6U PD=25.6U
M0 4 14 1 1 P W=24U L=4.8U AS=172.8e-12 AD=172.8e-12 PS=62.4U PD=62.4U
M2a 5 13 0 0 N W=12U L=12U AS=86.4e-12 AD=86.4e-12 PS=38.4U PD=38.4U
M2b 6 13 0 0 N W=12U L=12U AS=86.4e-12 AD=86.4e-12 PS=38.4U PD=38.4U
M3a 8 7 5 0 N W=48U L=4.8U AS=345.6e-12 AD=345.6e-12 PS=110.4U PD=110.4U
M3b 9 7 6 0 N W=48U L=4.8U AS=345.6e-12 AD=345.6e-12 PS=110.4U PD=110.4U
M4a 8 8 1 1 P W=24U L=12U AS=172.8e-12 AD=172.8e-12 PS=62.4U PD=62.4U
M4b 9 8 1 1 P W=24U L=12U AS=172.8e-12 AD=172.8e-12 PS=62.4U PD=62.4U
M5 12 9 1 1 P W=48U L=4.8U AS=345.6e-12 AD=345.6e-12 PS=110.4U PD=110.4U
M6 12 13 0 0 N W=24U L=12U AS=172.8e-12 AD=172.8e-12 PS=62.4U PD=62.4U
CM 12 6 0.4p
M7a 14 13 0 0 N W=12U L=12U AS=86.4e-12 AD=86.4e-12 PS=38.4U PD=38.4U
M8a 14 14 1 1 P W=24U L=4.8U AS=172.8e-12 AD=172.8e-12 PS=62.4U PD=62.4U
MI 13 13 0 0 N W=12U L=12U AS=86.4e-12 AD=86.4e-12 PS=38.4U PD=38.4U
ILOAD 1 13 DC 10U
Vcas 7 0 DC 2.00
.ENDS     $OPAMP
*
*
*** CLIPPING UNIT
*              in out
.SUBCKT CLIP   2  3
Eclip 3 0 VCVS 2 0 1 MAX=4.5 MIN=3.5
.ENDS     $CLIP
*
*
*** CURRENT CONVEYOR
*           in out vdd
.SUBCKT CC  2  3   1
Mcc1 4 2 1 1 P W=16U L=12U AS=115.2e-12 AD=115.2e-12 PS=46.4U PD=46.4U
Mcc2 3 4 2 1 P W=96U L=4.8U AS=691.2e-12 AD=691.2e-12 PS=206.4U PD=206.4U
MccI 4 6 0 0 N W=12U L=16U AS=86.4e-12 AD=86.4e-12 PS=38.4U PD=38.4U
Mccb 6 6 0 0 N W=12U L=16U AS=86.4e-12 AD=86.4e-12 PS=38.4U PD=38.4U
Icc 1 6 DC 3.6U
.ENDS     $CC
*
*
*** MULTIPLIER
*           in outl outr vw vwref
.SUBCKT MUL  4   5    6   2   3
M1 5 2 4 1 P W=4.8U L=27.2U AS=39.04e-12 AD=39.04e-12 PS=25.6u PD=25.6u
```

```
M2 6 3 4 1 P W=4.8U L=27.2U AS=39.04e-12 AD=39.04e-12 PS=25.6u PD=25.6u
.ENDS     $ MULTIPLIER
*
*
*** RESISTOR
*           in vref clock
.SUBCKT RES  2   4    99
ML 2 44 0 0 N W=4.8U L=6.4U AS=39e-12 AD=39e-12 PS=25.6U PD=25.6U
MSL 44 99 4 1 P W=2.4U L=2.4U AS=39e-12 AD=39e-12 PS=25.6U PD=25.6U
CL 44 0 0.1pF
.ENDS     $RESISTOR
*
*
*** COMPLETE CELL
*           vz  ul wul  u  wu  ur  wur  yl  wyl   y  wy  yr  wyr  vdd clock vref vwr vi vwi
.SUBCKT cel 4  21  22  17  18  25   26  23   24  16  20  27   28   1    99   3    2  29  30
XOA 11 12 15 1 OPAMP
XCLIP 15 16 CLIP
XRL 11 16 99 RES
XRR 12 3 99 RES
XCCL 13 11 1 CC
XCCR 14 12 1 CC
XFB 16 13 14 4 2 MUL
XOF 3 13 14 2 4 MUL
CL 11 0 0.1pF
CR 12 16 0.1pF
*
XU 17 13 14 18 2 MUL
XY 16 13 14 20 2 MUL
XUL 21 13 14 22 2 MUL
XYL 23 13 14 24 2 MUL
XUR 25 13 14 26 2 MUL
XYR 27 13 14 28 2 MUL
XI 29 13 14 30 2 MUL
.ENDS     $CEL
*
*****************
*    ANALYSIS: *
*****************
Vdd 1 0 DC 5.00
Vwref 2 0 DC 1.00
vref 3 0 DC 4.00
vz 4 0 PWL (0 1 10u 1 10.1u 1.8 20u 1.8 20.01u 1.40 50u 1.40)
Vclock 5 0 PWL (0 5 2u 5 2.01u 0 4u 0 4.01u 5 50u 5)
vi 6 0 DC 4.5
vrand 200 0 DC 3.5
vu1 11 0 DC 3.5
vu2 21 0 DC 4.5
vu3 31 0 DC 4.5
vu4 41 0 DC 3.5
vu5 51 0 DC 4.5
vu6 61 0 DC 4.5
vu7 71 0 DC 3.5
vu8 81 0 DC 4.5
vu9 91 0 DC 3.5
```

```
vA 100 0 PWL  (0 1 10u 1 10.01u 1    20u 1    20.01u 0.0 50u 0.0)
vB 101 0 PWL  (0 1 10u 1 10.01u 0.0  20u 0.0  20.01u 1   50u 1  )
vAL 102 0 PWL (0 1 10u 1 10.01u 1    20u 1    20.01u 0.5 50u 0.5)
vBL 103 0 PWL (0 1 10u 1 10.01u 1    20u 1    20.01u 1   50u 1  )
vAR 104 0 PWL (0 1 10u 1 10.01u 1    20u 1    20.01u 1.5 50u 1.5)
VBR 105 0 PWL (0 1 10u 1 10.01u 1    20u 1    20.01u 1   50u 1  )
VwI 106 0 PWL (0 1 10u 1 10.01u 1.50 20u 1.50 20.01u 1.4 50u 1.4)
*
* vz ul wul  u  wu ur wur yl wyl  y  wy yr wyr vdd clock vref vwr vi vw
X1 4 200  103 11 101 21 105 200  102 12 100 22 104  1   5    3    2   6 106 CEL
X2 4 11 103 21 101 31 105 12 102 22 100 32 104  1   5    3    2   6 106 CEL
X3 4 21 103 31 101 41 105 22 102 32 100 42 104  1   5    3    2   6 106 CEL
X4 4 31 103 41 101 51 105 32 102 42 100 52 104  1   5    3    2   6 106 CEL
X5 4 41 103 51 101 61 105 42 102 52 100 62 104  1   5    3    2   6 106 CEL
X6 4 51 103 61 101 71 105 52 102 62 100 72 104  1   5    3    2   6 106 CEL
X7 4 61 103 71 101 81 105 62 102 72 100 82 104  1   5    3    2   6 106 CEL
X8 4 71 103 81 101 91 105 72 102 82 100 92 104  1   5    3    2   6 106 CEL
X9 4 81 103 91 101 200 105 82 102 92 100 200 104 1  5    3    2   6 106 CEL
*****************
.TRAN 100n 100u 0u
*****************
.option post
.options list vntol=1e-12 abstol=1e-12 chgtol=1e-15 nomod
+ reltol=1e-4 dcon=1 absmos=1e-12 relmos=1e-4 pivot=13
+ gmin=1e-15 gmindc=1e-15 pivtol=1e-16 itl1=250 itl2=250
.END
```