The design and implementation of a switched current neural network

Eindhoven University of Technology
Faculty of Electrical Engineering
Electronic Circuit Design Group

Stan Ackermans Institute
Department of Information and Communication Technology

The Design and Implementation of a Switched Current Neural Network

Manjo Nijrolder

September 1995

Supervisors: dr. ir. J.A. Hegt
prof. dr. ir. W.M.G. van Bokhoven

Chapter 1

Abstract

A comparative study has been made to investigate hardware implementations of neural networks. The possibilities of switched current techniques are estimated and compared to other analog and mixed analog-digital techniques. The switched current technique can advantageuosly be used to implement synapses with respect to switched capacitor techniques due to its smaller occupied chip area. A system design of a hardware implementation of a perceptron neural net's forward path is made. The neural subsystem topologies are selected and dimensioned. The neural network is implemented in a $2.4\mu m$ N-well C-MOS technology.

Contents

Chapter 2

Introduction

Artificial neural networks have the potency to play an important role in the domain of signal processing in the future. They are used in areas where common signal processing systems that use algorithmic solutions are slow or show degraded performance. Well known application areas for neural networks are for instance pattern recognition and optimization problems. Another advantage is that neural networks do not need exact statistical knowledge of the problems to be solved, whereas most algorithmic signal processing systems do. Finally, large neural networks are fault tolerant, whereas algorithmic solutions are not.

These advantages are the result of the structural and functional properties of neural networks. Neural networks consist of a huge number of simple non-linear processing elements (neurons) that operate in parallel and are mutually connected. Due to this massive parallelism, a very high processing power is obtained. The fault tolerance of the neural network and the ability to tackle complicated problems is owed to the distribution of the signal processing over a huge number of parallel elements and to the non linear behavior of the neurons. Finally, the neural network's ability to learn from examples by means of a learning rule make neural networks attractive when statistical data of a problem is not known or too complex.

In the recent period, a lot of effort has been made to realize neural hardware and software in a variety of technologies. This report intents to investigate the possibilities of the switched current technique (SI) to implement neural networks, and to show the design and implementation of a switched current neural network. In this introduction, the switched current technique is introduced first. Second a number of neural hardware implementations in the CMOS technique is reviewed. Third, the possibilities of the switched current technique in neural hardware designs are estimated. Finally the switched current neural network design is introduced.

2.1 The switched current technique

The switched current technique is an analog sampled data technique that was first introduced by J.B. Hughes [30] in 1989. This technique uses the parasitic gate capacitance of a MOS transistor to implement a current memory. Because the parasitic capacitance is used, no linear capacitances are necessary, and a cheap standard digital CMOS process can be used for the implementation. By using a clocked version of the current memory, analog time discrete signal processing circuits can be implemented. Examples are delay lines, integrators, A/D and D/A converters, and even phase locked loops. The simplest switched current basic circuit is the second generation current copier cell [1]. The cell is depicted in figure 2.1.

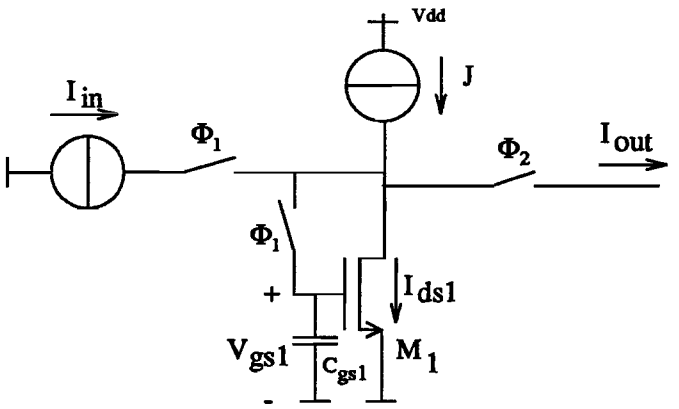

Figure 2.1: Second generation current copier cell.

The operation of the cell is as follows: In phase ϕ_1 the memory transistor M_1 is diode connected and the gate-source voltage (V_{gs1}) settles to a value that corresponds to a drain-source current $I_{ds1}^{\phi_1} = I_{in}^{\phi_1} + J$. In phase ϕ_2, the charge on the parasitic gate-source capacitance C_{gs1} is held. So the drain-source current is held ($I_{ds1}^{\phi_2} = I_{ds1}^{\phi_1}$). The output current therefore equals $I_{out}^{\phi_2} = -I_{in}^{\phi_1}$. The output current is a delayed and negated version of the input current.

In practical circuits, non-ideal behavior like channel length modulation and charge injection of the switches cause errors in the current transfer function of the memory cell. These errors can be minimized by dimensioning or compensated for by extra circuitry.

2.2 Review of neural hardware implementations

There are roughly three different groups of hardware for neural nets in the CMOS technology: Digital, analog, and mixed analog digital implementations.

- Digital implementations often use microcomputer structures that sequentially calculate the activation of a number of neurons. In this way, processing speed is traded off for flexibility, silicon area and precision [2]. Advantages of digital implementations are their programmability and accuracy. All known learning rules can be implemented on chip. Disadvantages are their use of large silicon areas with respect to analog implementations and their lack of exploiting the parallelism that is available in neural nets.

- Analog implementations use analog signal processing to calculate a neurons output signal. A large variety of analog processing techniques exist. Examples are current mode, transconductance mode, or time-sampled techniques as switched capacitor. When constant weights are used, and the neural net is not aimed at learning, analog implementations are advantageous because of their high processing speed and small chip area. Using standard chip technology, it is not yet possible to construct a non- volatile analog memory that memorizes an arbitrary value [3]. This property and the analog components limited accuracy are the main disadvantages of analog implementations. On-chip training is only possible with learning rules that tolerate the analog circuits inaccuracies such as stochastic training or weight perturbation.

- Mixed analog digital implementations use both digital and analog signal processing. They often apply digital weights storage and digital techniques to realize the learning algorithm and to make the neural net programmable. The calculation of a neurons output signal is usually implemented in an analog manner. An example is given in [4] that uses digital weights storing and off-chip training. Of course countless other combinations are possible.

In this report, only a limited number of implementations in analog and mixed analog digital techniques are taken into account. The reviewing is done in two steps: 1. Four important neural building blocks - Multiplier; Weights storage; Activation function; and Learning algorithm - are graded on their most significant design parameters. 2. These parameters - Accuracy; Micro-chip area; Supply voltage and power consumption; and Processing bandwidth - are compared.

2.2.1 Grading and comparing the neural hardware

In this section an incomplete, for this project relevant set of neural hardware is reviewed and compared. The grading of the neural building blocks is accounted for in appendix C.

2.2.2 Multiplier

The transconductance multiplier is mostly used in analog neural multipliers as is shown in table 2.1. It uses a relatively small number of transistors (4-19) and achieves intermediate accuracy (1- 5%) at high processing speeds. Chip area and power consumption of the shown chips using this multiplier are in the same order. Mixed analog digital implementations have accurate multipliers, but have lower processing speeds than transconductance multipliers. The area and number of transistors used by the multiplying digital to analog converters depends on the number of input bits. No accuracy data is available, but it is possible to realize low power realizations [4]. When only binary input signals and fixed binary weights are used, an area optimized multiplier is possible [16]. Some fixed weight multipliers are optimized on speed [10, 16].

Neural multipliers							
Reference	*Quadrants* -	*Type* -	*Area* [λ^2]	*Transistors* -	*Power* [Watt]	*Accuracy* %	*Speed* [MHz]
[4]	4	Multiplying A/D	5E3	30	10E-6	-	-
[9]	4	Gilbert	4.4E3	19	-	5	4.0
[15]	4	Transconductance	-	4	9E-4	-	3.3
[7]	4	Transconductance	2.8E3	4	3E-4	-	-
[17]	4	Gilbert	2.0E3	6	4E-4	-	-
[10]	4	Transistor ratios	-	9	-	2-5	20
[12]	4	Transconductance	-	5	-	1	-
[13]	4	Pulse width mod.	-	-	-	-	-
[11]	2	Synchrone pulse	2.1E3	-	3E-3	1	1
[16]	4	Capacitor ratios	18.3	0	-	1	10
[18, 19]	4	Multiplying A/D	8.6E3	13	2E-4	-	10

Table 2.1: Design parameters of neural multipliers

2.2.3 Weight storage

Analog weight storage is most common on the chips considered as can be seen in Table 2.2. Accuracies of up to 8 bits are used for realizing neural weight values. In this review, there is too little data available that deals with the weight storage chip area, but an advantageous combination of weight storage and multiplier implementation is used in [9] and [17]. This type of weight storage does not occupy extra chip area as the multipliers input transistor gate capacitance is used for weight storage. All chips that use analog weight storage, except for the "floating gates" chip [17] need refreshing circuitry, and external non-volatile digital memory. Digital weight storage [4] for analog neural net realizations is not often used, probably [3] because of the relatively large chip area required.

Weight storage				
Reference	Type	Analog or Digital	Accuracy [bit]	Area [λ^2]
[4]	Latch	D	7	5E3
[9]	Gate capacitance	A	-	0
[15]	Capacitor	A	-	-
[7]	-	-	-	3E3
[17]	Floating gate	A	4	-
[13]	Capacitor	A	8	-
[11]	Capacitor	A	-	-
[18, 19]	Capacitor	A	6	-

Table 2.2: Design parameters of weight storage mechanisms.

2.2.4 Activation function

The realization of a neural nets activation function depends on the required output signal. In case of binary outputs, the double inverter is most frequently used as is shown in Table 2.3. For analog valued output signals, transconductance circuitry is generally applied. Most perceptron implementations have a programmable gain sigmoid activation function, in order to be able to apply a varying number of synapses. The activation functions chip area is relatively less important than the synapses', because most neural networks use, in general, a smaller number of neurons than synapses.

Activation Functions				
Reference	Type	Area [λ^2]	Transistors	Gain programmable ?
[4]	channel length mod.	-	25	Y
[9]	Transconductance	4.8E3	13	Y
[15]	Transconductance	-	10	N
[7]	Transconductance	-	4	N
[17]	Transconductance	-	7+opamp	Y
[10]	V/I converter	-	2	N
[12]	Current mode PWL	-	12 or 18	N
[13]	-	-	-	N
[11]	Double inverter	-	4	N
[16]	Double inverter	-	5	N
[18, 19]	A/D converter	-	-	Y

Table 2.3: Design parameters of activation functions.

2.2.5 General system design parameters

Table 2.4 shows that current mode circuits require relatively small supply voltages, and that high speed current mode applications are possible. Only mixed analog digital implementations have been developed that use on-chip learning. The Kohonen Learning rule is used on those chips. The kohonen learning algorithm is a relatively simple rule, and it uses locally available signals. Therefore it might be the most suitable algorithm to be implemented on chip. Most analog realizations are adapted to one type of neural network, it thus can be stated that purely analog circuitry is less flexible than mixed analog digital circuits [17, 18, 19].

General parameters						
Reference	Type	Technology λ [μm]	Speed [MHz]	Voltage [V]	Power [Watt/synapse]	Learning
[4]	Perceptron	2.0	-	0;10	10E-6	Off-chip
[9]	Perceptron	2.0	4.0	-5;5	-	Off-chip
[15]	Cellular	1.5	3.3	-5;5	9E-4	NO
[7]	Hopfield	2.0	-	0;5	3E-4	Off-chip
[17]	Hopf./Perc.	1.0	-	-5;5	4E-4	Off-chip
[10]	Perceptron	1.0	20	0;5	-	NO
[12]	Chaotic	1.6	-	0;3	-	Off-chip
[13]	Kohonen	1.6	-	0;5	-	Kohonen
[11]	Kohonen	1.2	1.0	-6;6	3E-3	Kohonen
[16]	Perceptron	3.0	10	0;5	-	NO
[18, 19]	Hopf./Perc.	0.9	10	0;5	2E-4	Off-chip

Table 2.4: General design parameters of neural hardware.

2.3 Possibilities of the switched current technique

In order to estimate the possibilities of switched current techniques in the domain of neural hardware implementations, a related technique has to be found, in which neural nets have been implemented. The SI technique is suitable for mixed analog digital circuitry as for instance the Switched Capacitor (SC) technique is. So evaluating the possibilities of SI techniques, it can be compared with the SC technique. Programmability of a circuit, available in SC circuits, is also possible in the SI technique. The biggest advantages of SI with respect to SC are its reduced chip area and cheap production processes [20]. In most neural networks, a large number of synapses is used, taking the bulk of the total chip area. Area reduction of the neural multiplier and weight storage by using SI instead of SC would therefore be highly favorable. The design of activation functions allows a lot more freedom of design, so the major benefit of SI being its reduced chip area is not significant. On chip implementation of learning algorithms requires highly accurate circuits such as offered by SC techniques. In [20] it is stated that in SI circuitry, matching accuracy between circuit components is easier to achieve than in SC circuitry. The attainable signal to noise ratio however, is higher in SC than in SI. It is therefore not possible to state whether SI is suited to implement on-chip learning algorithms.

2.4 The switched current neural network design

The switched current neural network design has to exploit the advantages of the switched current technique. The most important advantages of switched current are its limited chip area and power consumption. These are important design issues in synapse design. Hence, the design will focus on the synapse (multiplier and weight storage). In order to obtain a functional neural network, a perceptron neural network is chosen for because it is most common.
A number of system design parameters can be extracted from the reviewed chips. The required weight storage accuracy is set to 8 bits (table 2.2). The power supply is set to $5V$, and the operating speed is chosen to be 1 MHz which is a common value for switched capacitor implementations [11].

Chapter 3

System design of a switched current neural net

3.1 Designing the forward path of a neural net

The implementation of the forward path subsystem as shown in figure 3.1 can be a first step in realizing a switched current neural network. The forward path subsystem consists of synapses and the activation function. A synapse consists of a multiplier and a weight memory. It is dealt with in section 3.2 to 3.3. Section 3.4 deals with the system design of the activation function.

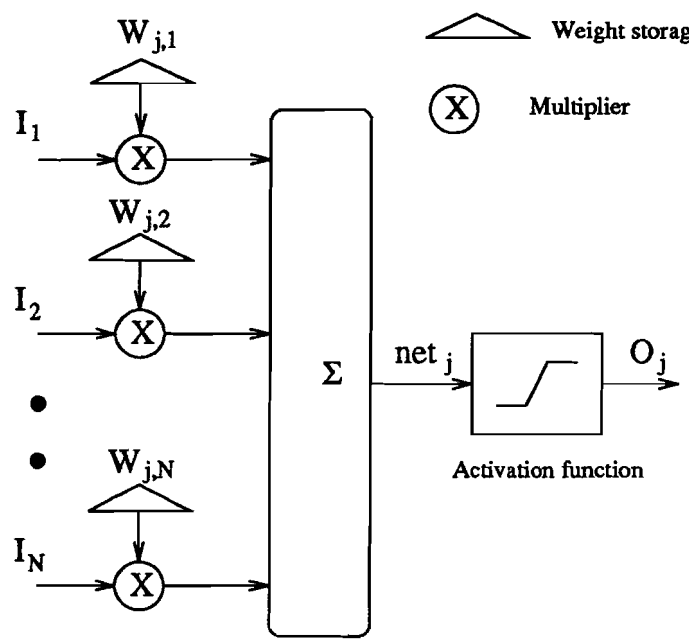

Figure 3.1: A neural net forward path subsystem.

The mathematical function implemented in a forward path, as depicted in figure 3.1, is given in formula 3.1

$$O_j = S(net_j) \tag{3.1}$$

$$net_j = \sum_{n=1}^{N} W_{j,n} I_n$$

The weight $W_{j,1}$ is a bias weight, with $I_1 = 1$. For large neural networks, the number N is large. The chip area and power consumption of the synapse are therefore important design issues in case only the forward path is implemented.

The activation function is nonlinear. A variety of activation function shapes exist. Examples are hard limiter, sigmoid and hyperbolic tangent. Only one activation function is required for N synapses, so more freedom with respect to chip area and power consumption is allowed in the implementation of the activation function.

3.2 Synapse design

In an electronic neural implementation, the weights $W_{j,n}$ are signed and limited. $-lw_{lim} < W_{j,n} < hw_{lim}$, with $lw_{lim}, hw_{lim} > 0$. The input signals I_n are limited, and may be signed, denoted as I_n^s or unsigned (i_n^u). $li_{lim} < I_n^u < hi_{lim}$, with $li_{lim}, hi_{lim} > 0$. The unsigned input signal can be written as a shifted (by I_c) signed input signal:

$$I_n^s = I_n^u + I_c \tag{3.2}$$

The resulting sum of unsigned input signals (net_j^u) equals:

$$\begin{aligned} net_j^u &= \sum_{n=1}^{N} W_{j,n} I_n^u \\ \Rightarrow net_j &= net_j^u + \sum_{n=0}^{N} W_{j,n} I_c \end{aligned} \tag{3.3}$$

So, the resulting sum of unsigned input signals net_j^u has to be shifted by an amount of $\sum_{n=0}^{N} W_{j,n} i_c$ with respect to the resulting sum for signed input signals. This shift can be accounted for by the bias weight $W_{j,0}$. This means that a two quadrant multiplier is sufficient for the implementation of the forward path of a neural net. In section 3.3, a choice will be made between a four and a two quadrant multiplier.

3.2.1 Multiplier input and output signal representations

The multiplier has two input signals (figure 3.1), the weight from the local weight memory and the input signal from the former layer or from the outside. It is appropriate to implement distributed signals as voltages. The input signal coming from the former layer is therefore represented by a voltage.

The weight signal is a signal that only has to be available locally, so more freedom is allowed in the representation of the weight signal. Switched current techniques are appropriate to memorize currents. It is therefor suitable to realize a weight memory by means of switched current memory cells. A current is appropriate to represent a weight.

The multiplier output signals have to be summed in a summing node. The summing can be realized easily by using Kirchhoff's current law. The multipliers output signal therefore has to be a current.

3.2.2 Multiplying a weight current with an input voltage to obtain an output current

In this neural implementation, the multipliers occupied chip area is an important property. Hence a multiplying principle that does not require V-I or I-V interfacing is more appropriate than principles that do. Multipliers using pulse stream techniques belong to this class of multipliers.

- A very simple four quadrant multiplier is shown in figure 3.2.

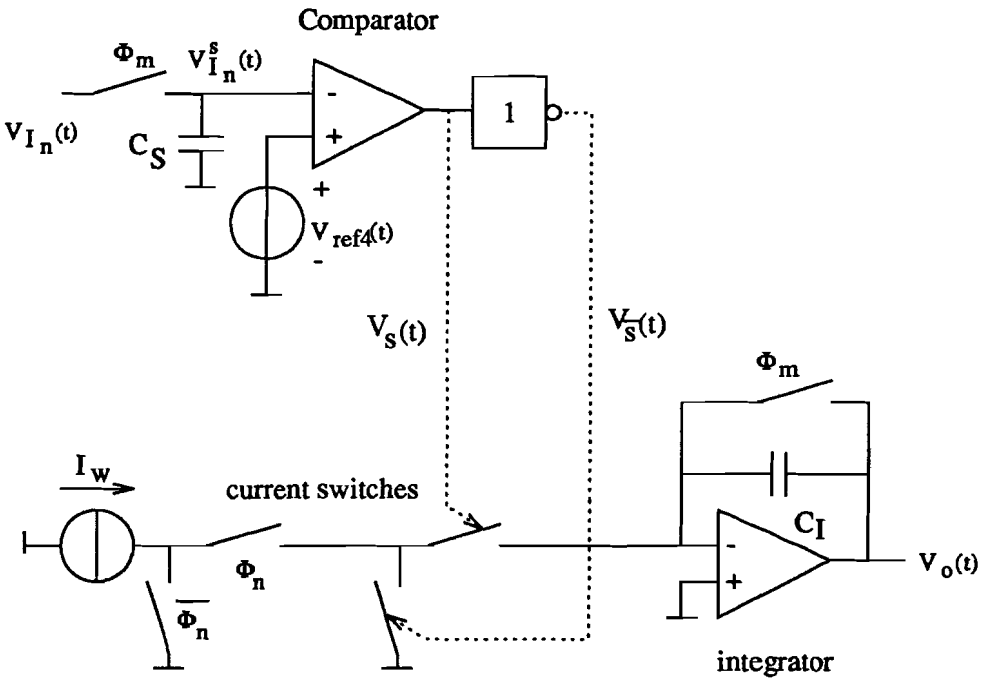

Figure 3.2: Four quadrant multiplier with pulse stream generator.

This multiplier operates in phase ϕ_n. The result is available at the end of this phase. In phase ϕ_m, the multiplier is reset. In this principle, a binary valued switching voltage ($V_s(t)$) is generated by an immediate comparison of a sampled analog input signal ($V^s_{I_n}(t)$) and a reference signal ($V_{ref4}(t)$). This switching voltage controls a number of current switches. The input weight (I_w) current is directly applied to the input of an integrator. The multiplication result ($V_o(t)$) can be obtained by integration of the weight current.

In neural networks, an array of multipliers (1..N) is necessary to calculate a sum of products. As integrating and summing are linear operations, they can be interchanged. This can be used to calculate a sum of products more efficiently. Instead of summing the integrator output voltages, the integrator input currents are summed. Then, only one integrator is necessary for N products. The signal shapes involved are shown in figure 3.3.

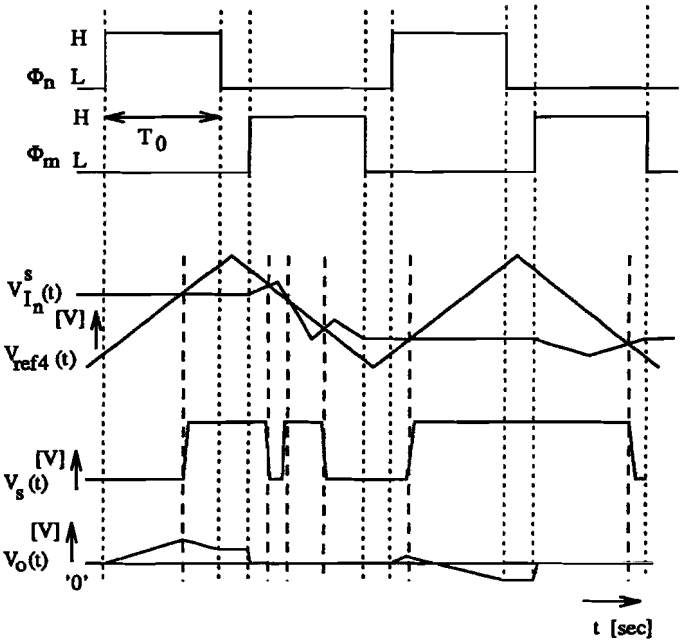

Figure 3.3: Four quadrant pulse stream multiplier signal shapes.

- The same multiplying principle can be used in a two quadrant multiplier (figure 3.4).

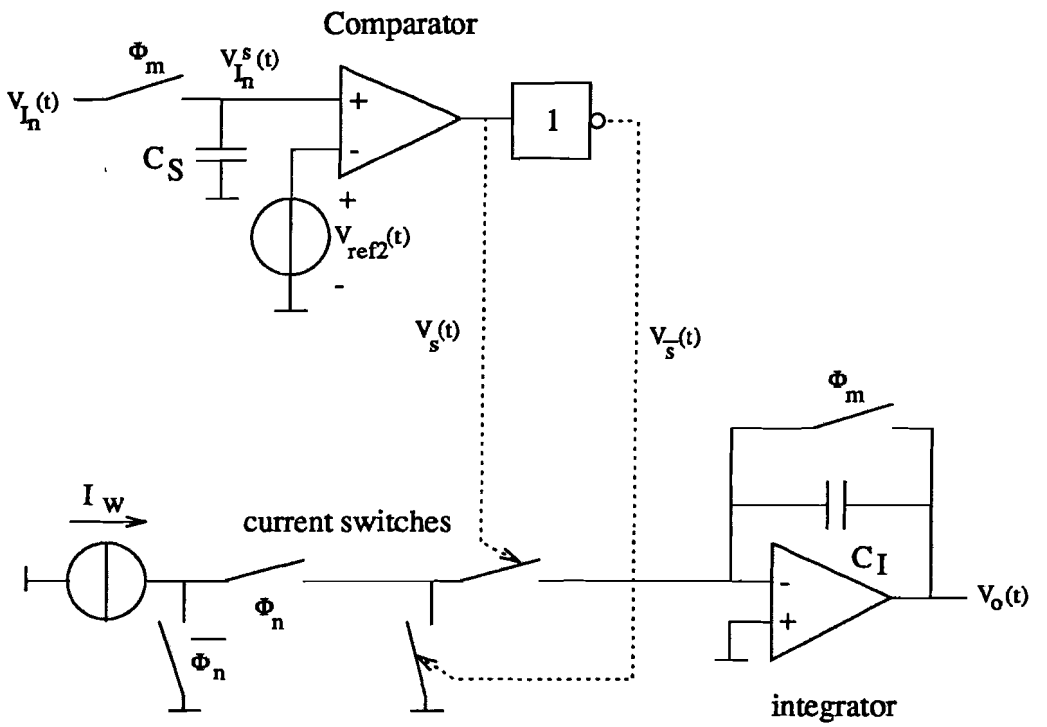

Figure 3.4: Two quadrant pulse stream multiplier with pulse width modulator.

This multiplier also operates in phase ϕ_n. The result ($V_o(t)$) is available at the end of this phase. In phase ϕ_m, the multiplier is reset again. In this principle, a binary valued switching voltage ($V_s(t)$) is generated by an immediate comparison of a sampled analog input signal ($V_{I_n}^s(t)$) and a reference signal ($V_{ref2}(t)$). The reference voltage is dimensioned such a way that the binary switching voltage is pulse width modulated (figure 3.5). The output signal ($V_o(t)$) is the integrated pulse width modulated weight current. Again, interchanging the integrating and summing give an efficient way of calculating a sum of products. The signal shapes of this multiplier are depicted in figure 3.5.

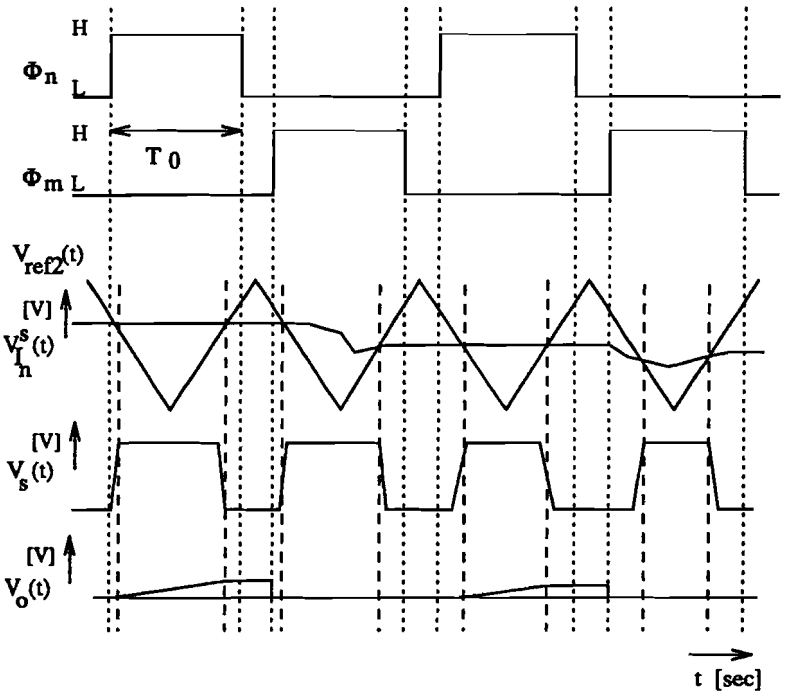

Figure 3.5: Two quadrant pulse width multiplier signal shapes.

This two quadrant multiplying principle is verified in appendix D. Figure 3.5 shows that the two quadrant multiplier uses a reference signal ($V_{ref2}(t)$) with a two times higher fundamental frequency than the four quadrant multiplier.

3.3 Two or four quadrant multiplier

In this section, some properties of the presented multipliers are summarized, and a choice made whether the two or the four quadrant multiplier is implemented. Some advantages of using a two quadrant multiplier are:

- From figures 3.2 and 3.4 it is obvious that the two quadrant multiplier uses less hardware than the four quadrant multiplier.

Some drawbacks of using the two quadrant multiplier are:

- A two quadrant multiplier is not suitable for the implementation of learning algorithms (like back propagation) that need four quadrant multipliers.

- The fundamental frequency of the reference signal of the two quadrant multiplier is two times higher than the four quadrant multiplier's.

It is decided to implement a two quadrant multiplier because it requires less hardware, and this project only focusses on the feed forward neural net.

3.4 Neuron design

The neuron in the forward path of a neural net implements the activation (O_j) of the summing result (net_j): $O_j = S(net_j)$. The network uses two quadrant multipliers, so the neurons output signal has to be unsigned. So an unsigned acitvation function like a sigmoid can be implemented. In order to apply a varying number of synapses, the gain of the neuron must be adaptable. In neuron design, the consumed power and occupied chip area are less important than the processing speed.

3.4.1 Neuron signal representations

The neurons input signal is a voltage ($V_o(t)$), generated by the integration device. The output signal has to be suitable for comparison with a reference voltage ($V_{ref2}(t)$) for the pulse width modulator for the multipliers source encoder. Hence a voltage would be an appropriate neuron output signal.

3.4.2 Activation function principle

In designing the activation functions principle, the processing speed is the most important. Two principles will be summarized here.

- A possible implementation uses the reference voltage of the pulse width modulator to realize an activation function [21]. This principle is shown in figure 3.6.

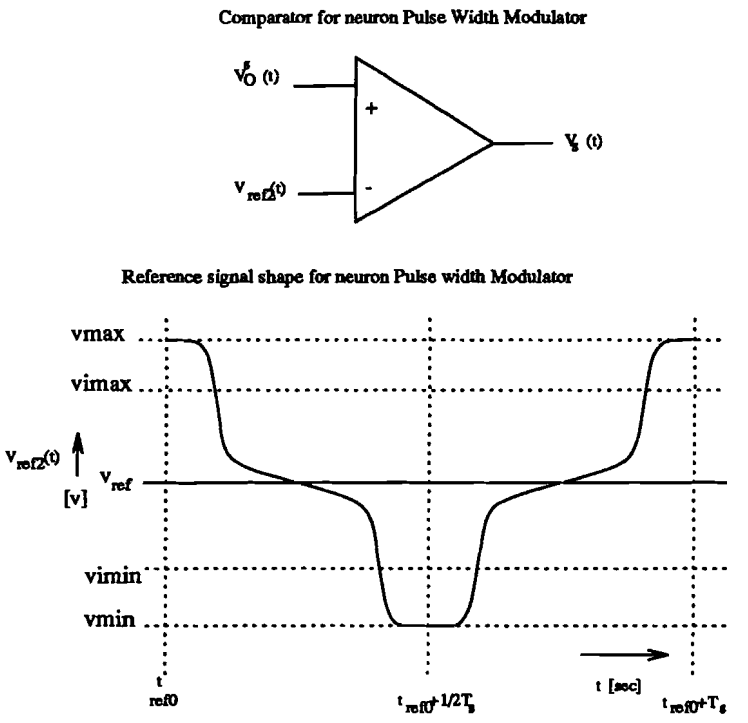

Figure 3.6: Integration of neuron and pulse width modulator

This principle costs virtualy no processing time. The gain of the activation function can by varied by changing the slope of the reference voltage. A drawback is that countermeasures have to be taken to prevent the integration devices output signal ($V_o(t)$) from exceeding the input signal limits v_{imin} and v_{imax}. Another drawback may be the generation of the reference signal, which may cost some extra hardware.

- A voltage controled non-linear voltage source (figure 3.7). This may be the most straight forward way to realize an activation function. This solution inherently costs more processing time than the first principle. Advantages of this principle are: A. No preprocessing of the integration devices output signal $V_o(t)$ is required. B. Hardware to vary the gain of the neuron is quite easy to implement in this principle. Disadvantages are: A. The processing time of the neuron. B. The hardware is not shared with the pulse width modulator like in the first principle. C. The amount of hardware necessary for the voltage controled voltage source has to be implemented for each neuron, whereas the hardware for the generation of the integrated pulse width modulator neuron only has to be implemented once for the whole chip.

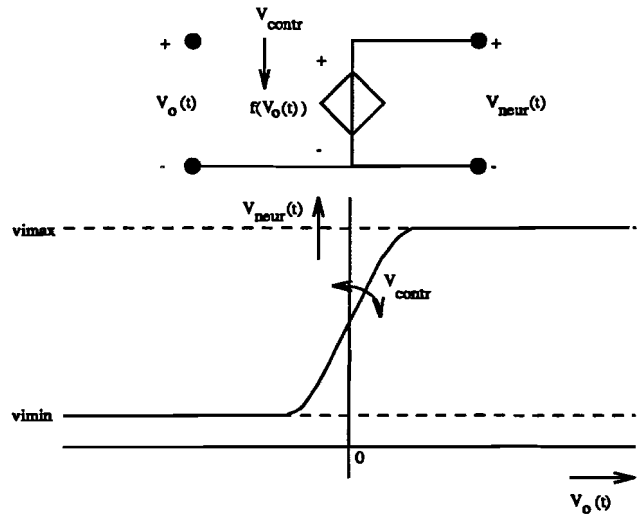

Figure 3.7: Voltage controled voltage source neuron.

The extra processing time needed for the voltage controled voltage source and the extra amount of hardware needed for each neuron makes the integrated neuron and pulse width modulator more appropriate in this network.

Chapter 4

From design principles to topology selection

The realization principles that were chosen in chapter 3 have to be worked out to topologies of neural hardware. The subsystems: current memory, current switches, integrator and comparator will be considered in this chapter. Again, for the implementation of the current memory, and the current switches, the occupied chip area, and the consumed power are the most important design issues.

4.1 The current memory

The current memory provides the weight current I_w for the neural net. The current memory must have the following properties:

- The weight current must be signed.
- The weight current must be maintained as long as possible within accuracy limits as they will be described in chapter 5.
- The weight memory must have a relative high output impedance.
- The weight current must be monotonic with respect to the weight.

The first item states that the weight current must be signed. This can be implemented by either a biased current memory [23] (figure 4.1), or a double unsigned current memory [25]. Because the chip area is an important property, a biased current memory is used.

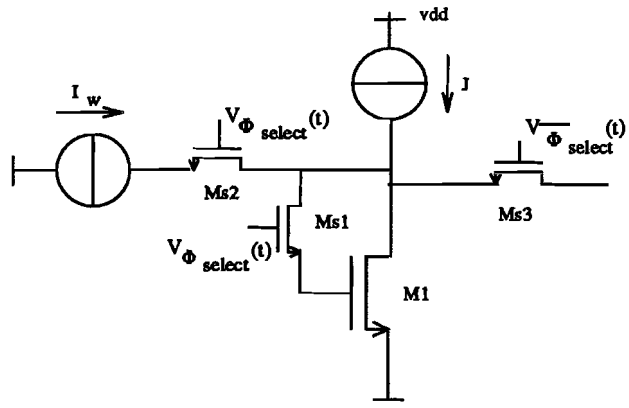

Figure 4.1: A biased current memory.

The second item states that the weight has to be memorized as long as possible. A certain weight refresh period T_r is needed to keep the weight within the required accuracy limits. This refresh period T_r is determined by the drift of the weight current and the required accuracy. Drifting is caused by leakage currents of the switching transistors. The weight current of a differential weight memory [24] is in first order approximation insensitive for leakage currents (figure 4.2). This means that using twice the capacity of the normal biased memory cell, the weight can be held for one or two orders longer. The differential memory cell requires common mode feed-back circuitry to match the current sources (J ; $2J$). This circuitry is not shown in figure 4.2.

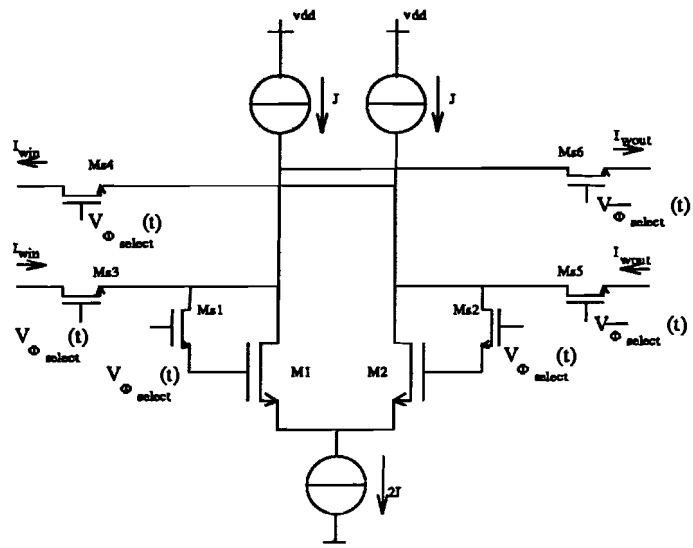

Figure 4.2: A N-type differential current memory.

The refresh period can naturally be extended by enlarging the memory transistors gate capacitance. A large gate capacitance means large transistors. A compromise must be found here to keep the occupied chip area limited and to keep the memory cell fast enough to be initialized and refreshed in one clock cycle (T_s).

The third item deals with the cell's output impedance. The output impedance of the weight memory must be high with respect to the impedance of the 'virtual ground' of the integrator. This impedance ratio scales the synapses output currents. The impedance ratio can be increased by increasing the output impedance of the current memory, or by reducing the input impedance of the integration device. A common method of increasing the output impedance is to use a cascoded configuration. This requires extra transistors and biasing circuitry for each synapse. The input impedance of the integrator can be reduced by increasing the integrator operational amplifier's gain. This only costs circuitry for the operational amplifier, so only once for a large number of synapses. The current memory will not be cascoded for this reason.

The last item treats the monotonicity of the weight current with respect to the weight. Deflection of monotonicity can be caused by the non-idealities of the current memory. The main error sources of the current memory are: Charge injection of the switching transistors, settling errors during refreshing, impedance ratio errors and noise. These errors do not deflect the weight current from monotonic behavior, (appendix E) except for the noise error. No extra circuitry is required to compensate for these errors, but at dimensioning, these errors are minimized. The noise error is reduced by taking the most appropriate topology (P-MOS cell), and by optimizing the dimensions for the signal to noise ratio. The resulting topologies are given in figure 4.3. By deleting one of the bias current sources (J) of figure 4.2, no common mode feedback circuitry is needed.

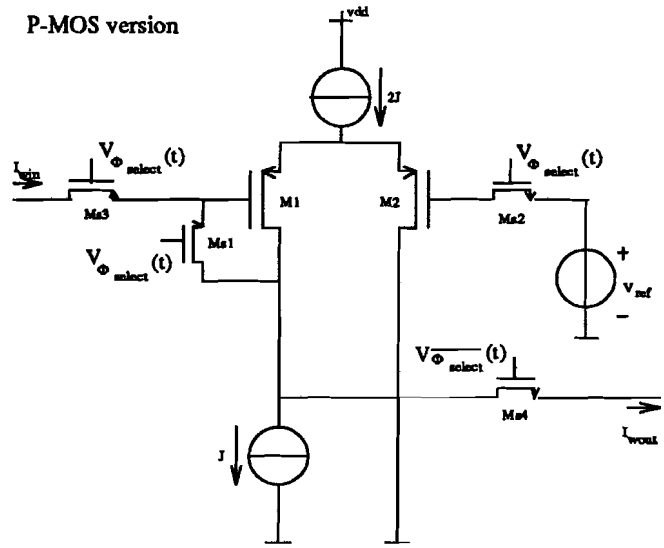

Figure 4.3: The resulting current memory circuit.

4.2 The current switches

The current switches are used to connect a high impedance node - the current memory - to a low impedance node - the integration device input terminal. Some switches are shown in figure 4.4. The current switches must have the following properties:

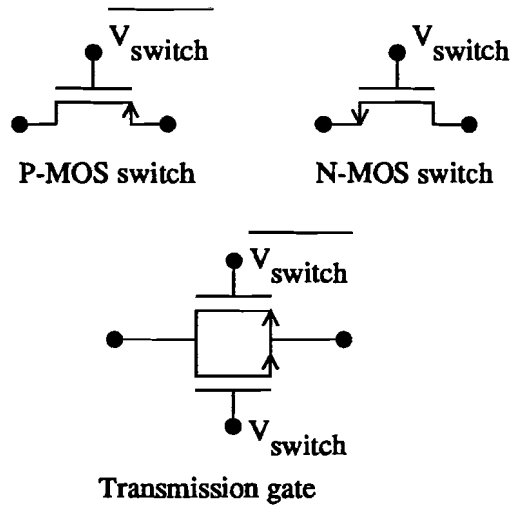

Figure 4.4: CMOS switches.

- The on-resistance of the switches must be low.
- The charge injection of the current switches must be low.
- The leakage current of the current switches must be low.
- The errors involved with the current switches may not deflect the weight current from monoticity.

The transmission gate switch in figure 4.4 is appropriate if the voltage of the terminals varies over a large range. The single N-MOS and P-MOS switches can be used in case of a limited voltage range. The voltage of the low impedance terminal is constant and determined by the integration device. As the switches are used for a limited voltage range, a single MOS switch can be used. The on-resistance of the P-MOS switch is, at the same aspect ratio, higher than the on-resistance of the N-MOS switch. The body effect in a N-well process is stronger for the P-MOS than for the N-MOS, so N-MOS switches operate at a larger voltage range than P-MOS switches. These both factors make N-MOS switches more appropriate than P-MOS switches. The errors involved with the current switches are: non-zero on-resistance, leakage currents, charge injection at switching instances and noise. These errors do not deflect the weight current from monotonity, except for the noise error. The noise error can be minimized by means of dimensioning.

4.3 The integrator

The integrator integrates the synapse output current to a sum of products result. The integrator must have the following properties:

- The slew rate of the integrator must be sufficiently high.
- The gain-bandwidth product of the integrator must be sufficiently high.
- The integrator must have a large output voltage swing.
- The integrator must be prevented from clipping.

- The virtual ground nodes input impedance must be low.

The slew rate of the integrators operational amplifier must be sufficiently high, to prevent distortion of the multiplication result as is derived in appendix F. This requirement can be achieved by dimensioning the opamp, by reducing the maximum weight current and by enlarging the integration capacitance.
The gain-bandwidth product of the operational amplifier must be high enough to keep the voltage of the virtual ground within its input range. This requirement is derived in section F.1.3.
The integrators output voltage must be prevented from clipping. This is necessary in order to keep the virtual ground nodes input resistance low. A way of doing this is to clamp the integrators output voltage (figure 6.6).

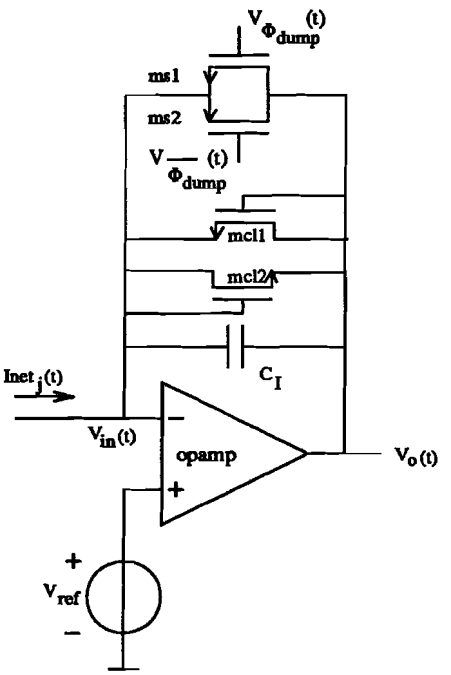

Figure 4.5: The integrator.

The input impedance at the virtual ground node can be kept low by maintaining the gain of the opamp sufficiently high. Again this can be achieved by dimensioning.
The integrator must be reset after each integration phase. The reset switch must operate at a large voltage range. A transmission gate is appropriate here.
The opamp neither drives a large capacitive nor a low resistive load. Therefore an opamp output buffer is not needed.
The opamp has to operate at a fixed input common mode level, therefore no requirements to enlarge the input common mode range are needed.
A large output swing is required, and can easily be achieved by using an inverter output stage.
The gain of the opamp can be enlarged by cascoding or cascading. As an extra inverter stage already is inserted, the gain of the opamp is already relative high, so no extra cascoding is required. Hence the opamp of figure 4.6 is proposed.

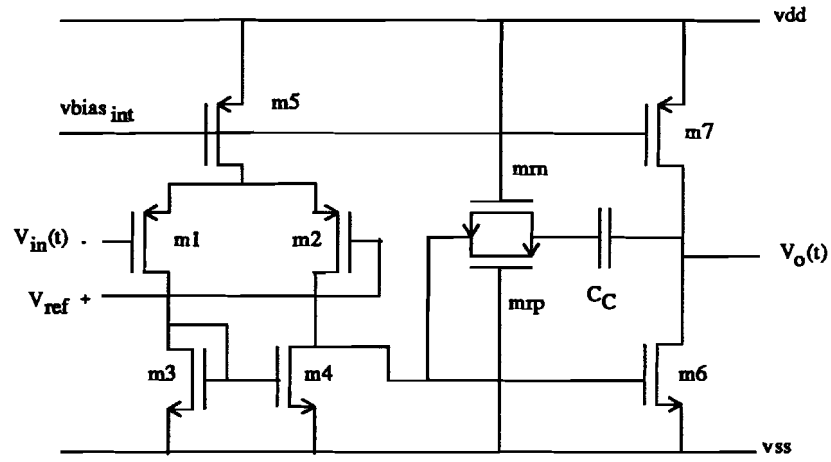

Figure 4.6: The proposed opamp topology.

4.4 The comparator

The comparator compares an input signal ($V_{I_n}^s(t)$) with a reference voltage ($V_{ref2}(t)$). Some important design objectives for the comparator are:

- The comparator must have a large input common mode range.
- The comparator's slew rate must be sufficiently high.
- The comparator's offset must be low.
- The comparator must be able to drive a relatively large output capacitance.

The first item treats the input common mode range. The comparators input signals ($V_{I_n^s(t)} and V_{ref2}(t)$) occupy a large voltage range. The maximum input range for hidden layers is determined by the integrators output clamping voltage. The proposed topology is depicted in figure 4.7.

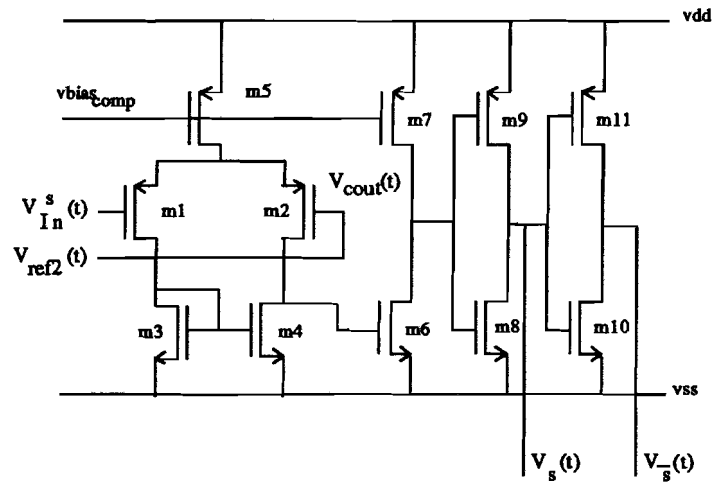

Figure 4.7: The proposed comparator topology.

The common mode input range of this comparator is limited for high common mode voltages. For low common mode voltages, more voltage space is available. So by decreasing the reference voltage of the integrator, the available voltage space is used more efficiently without using extra circuitry.

The second item deals with the comparators slew rate. The most critical situation occurs at minimum pulse width. The slew rate can be adapted by means of the bias currents (I_{M5} and I_{M7}). The required slew rate is derived in appendix F.

The third item deals with the comparators offset voltage. Offset voltages result in extra offset of the pulse width modulated input signal. Offsets do not affect the monotonicity of the input signals, so no extra circuitry is required to correct for this offset. The comparators offset is minimized by dimensioning of the current mirror.

The last item deals with the comparators driving capability. The comparators load driving capability can be adapted by means of buffering. The propagation delay of this comparator may be large with respect to the systems clock period. So, in cascaded layers of neurons, the neurons output signal is delayed with respect to the system clocks (ϕ_n; ϕ_m). Problems resulting from this propagation delay should be dealt with on a system level, rather then trying to reduce it by applying extra chip area and power. A possible solution is to generate the system clocks (ϕ_n; ϕ_m) from the reference signal $V_{ref2}(t)$. In this way, the clock is delayed by approximately the same amount as the pulse stream signal ($V_s(t)$) is delayed by the comparator.

Chapter 5

Dimensioning the neural hardware

The dimensions of the topologies of chapter 4 have to be chosen. In order to come up with a rational dimensioned circuit, at first, the boundary conditions will be determined.

5.1 The boundary conditions for dimensioning the neural hardware

The boundary conditions of the design have to be established. This involves the process parameters and system variables.

5.1.1 The process parameters

The Mietec 2.4 μm N-well C-MOS process is used to implement this hardware. The level 2 HSPICE [28] parameters are given in appendix B. This process supports two metal layers and two polysilicon layers. The polysilicon layers allow linear poly1-poly2 capacitors. These process parameters are extracted in august 1992, so a significant deviation in the real parameters may occur. In order to obtain sufficient mirroring accuracy between transistors, the smallest possible dimensions should not be used. Therefore a minimum dimension of 4.8 μm is applied at critical elements.

5.1.2 The system variables

The system variables concern a range of quantities such as the voltages, currents, timing, and accuracy definition. The proposed quantities are summarized below.

Quantity	amount	Unit	description
vdd	5	[V]	supply voltage
vss	0	[V]	supply voltage
Vref	2	[V]	reference voltage
V_{clamp}	1.3	[V]	integrator clamping voltage
$V_{act_{max}}$	0.5	[V]	activation function's extreme input value
I_{wmax}	5	[μA]	maximum weight current
T_0	400	[$nsec$]	system clock high period
T_s	500	[$nsec$]	pulse width reference signal period
T_{select}	500	[μsec]	memory cell's refresh time
T_r	0.5	[$msec$]	memory cell's refresh period
T_{PWmax}	400	[$nsec$]	maximum multiplying pulse width

T_{PWmin}	40	$[nsec]$	minimum multiplying pulse width
ϵ_s	$< 2^{-8}$	-	refreshing error
E_{in}	$< 2^{-8}$	-	input weight accuracy
N	8	-	number of synapses per neuron

Some of the quantities ($V_{ref}; T_{PWmin}; I_{wmax}$) are obtained in an iterative way throughout the designing process. Other quantities are obtained from literature (E_{in}). The settling error (ϵ_s) is chosen at the same accuracy as the input weight error (E_{in}). This implies that the worst case weight storage error is $\epsilon_s + E_{in}$. Some parasitic capacitances are not known on forehand. Therefore some estimates are made that influence the system variables. Especially the dynamic behavior of the circuitry is influenced by the parasitic capacitances.

5.2 Dimensioning the memory cell

The power consumption and chip area are the most important design issues while dimensioning the memory cell. The first step is to decide on the memory cell type. The area of the synapse is to a large extend determined by the memory cell's differential pair. The P-type differential pair involves less noise and a higher output impedance than the N-type differential pair. So, in order to achieve a comparable performance with a N-type cell, more chip area has to be used. For this reason, the P-type memory cell is used as depicted in figure 5.1.

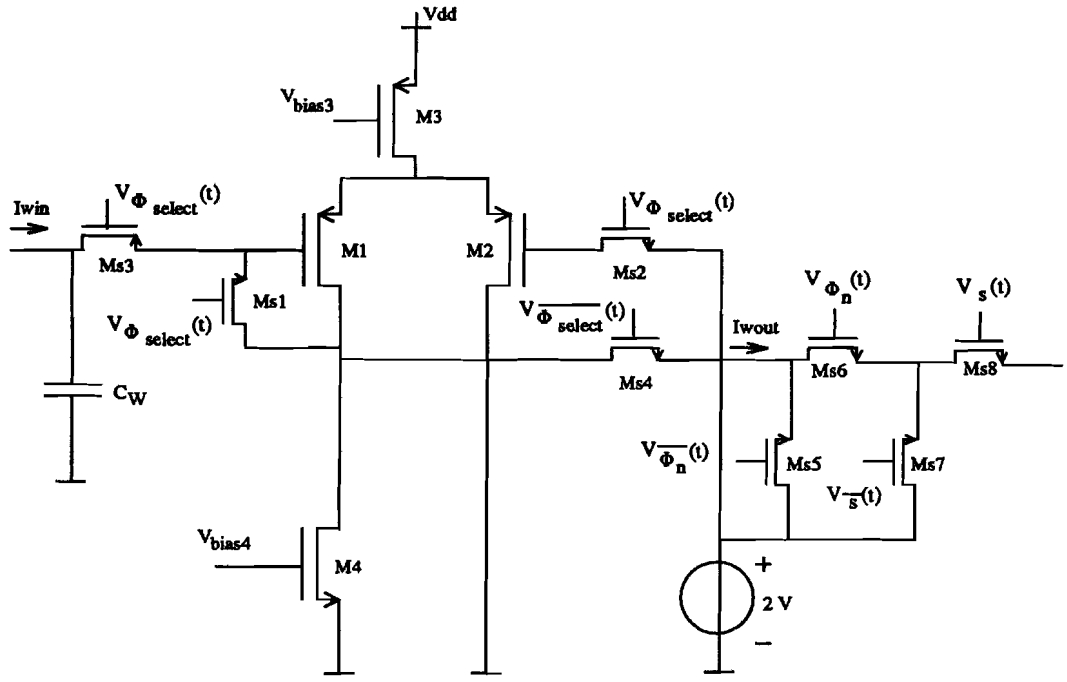

Figure 5.1: P-type memory cell with current steering switches.

A rational compromise has to be found for the dimensions of the transistors of the P-type memory cell, based on the following items:

- Bias currents The maximum weight current ($I_{wmax} = 5\mu A$) determines I_{dM3} and I_{dM4}: ($I_{dM3} = -10\mu A$; $I_{dM4} = 5\mu A$).

- DC-operation In the analysis of appendix E, the transistors (M1-M4) are assumed to be in saturation. A common mode output range ($CMOR = 1V$) is required to permit modulation off the virtual earth node of the integrator.

- Settling behavior An estimate is made concerning the wiring capacitance ($C_w = 0.5pF$). This capacitance determines the dynamical behavior of the cell, together with the gate-source capacitances (C_{gs}) of $M1$ and $M2$. A settling error of less than 2^{-8} is required. The settling error is calculated in appendix E, formula E.13. When $C_w \gg C_{gs}$, the relative error limit value equals $\epsilon_s < 2e^{-T_r/\tau}$. In case of linear settling, the constraint for the cell's dominant time constant (τ) becomes $\tau < \frac{-T_r}{ln(2^{-8}/2)}$. Monotonic settling is easily achieved with this type of memory cell, provided the on resistance of the switches is low enough. By taking minimum sized N-MOS switches, this condition is satisfied.

- Refresh cycle The refresh cycle depends on the reverse biased diode leakage current and the charge injection through M1's drain modulation. The injected charge also depends of the overlap capacitances. They can be calculated from the layout, so these effects are simulated in section 6.0.2. In general it can be stated that, the larger the gate-source capacitances, the longer the memorized weight current is maintained within its accuracy limits.

- Noise and mismatch Both noise and mismatch are reduced by taking larger transistors (WL large).

- Output conductance The output conductance of the memory cell can be made smaller by using longer transistors.

- Switches The switches (Ms1..Ms8) are minimum sized: $W = L = 2.4\,\mu$. In this way, the charge injection and the capacitive loading of the switches's controlling circuitry is minimized.

The dimensions of the memory cell are given in table 5.1. The dimensions are chosen through hand calculations (appendix G) and HSPICE simulations.

Table 5.1: Dimensions differential memory cell

Differential memory cell			
Transistor	*W*	*L*	*unit*
M_1	42	7.2	μm
M_2	42	7.2	μm
M_3	24	4.8	μm
M_4	4.8	24	μm

5.3 Dimensioning the integrator

The complete integrator scheme with output sampling circuitry is given in figure 5.2.

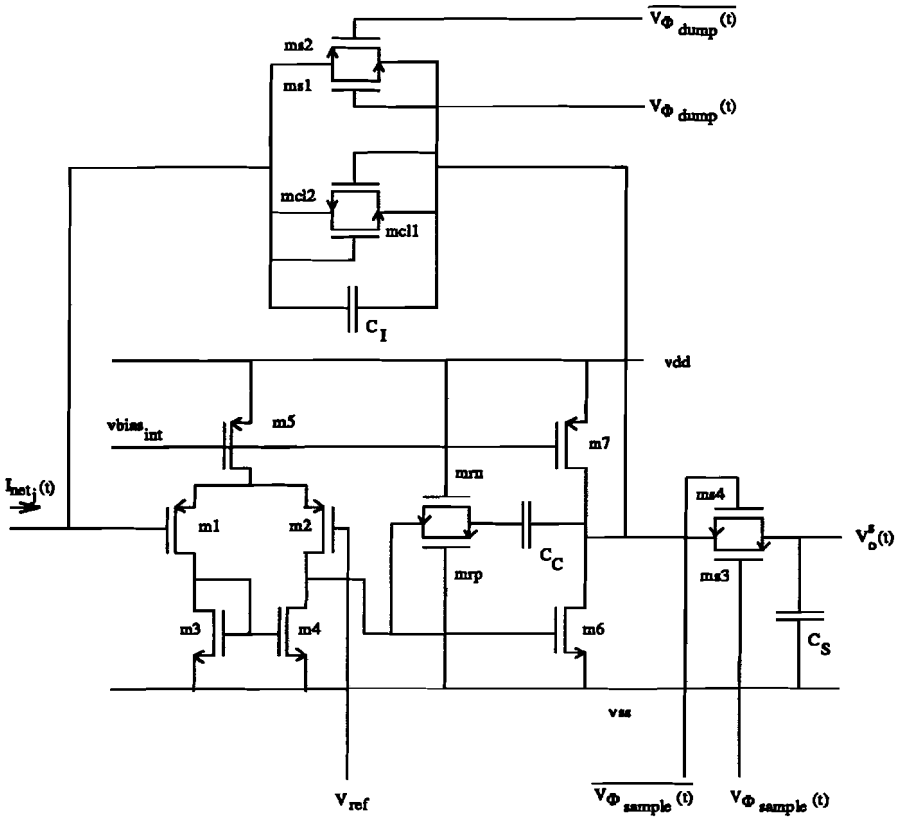

Figure 5.2: Complete integrator scheme with output sampling circuitry.

The power consumption and settling behavior are the most important design issues. The following items are taken into account:

- The integration capacitor. The integration capacitor (C_I) depends on the number of synapses (N), the maximum multiplying pulse width (T_{PWmax}), the clamping voltage (V_{clamp}) and the activation function's extreme input value ($V_{act_{max}}$), as is shown in formula F.5.
- DC-operation The transistors of the integrator's opamp are assumed to be in saturation.
- The output stage current. The output stage current (I_{dM7}) is dependent on the required slew rate (SR) and the output stages load capacitance.
- The settling behavior. For the integrator, as for the memory cell, monotonic settling is required. It can be achieved by keeping the non-dominant poles $4(A_0 + 1)$ from the dominant pole. In this way, the closed loop system of the integrator or the unity-gain opamp only contains real poles.
- Noise and mismatch. In order to reduce the noise and the mismatch, the transistors have to be taken as large as possible. The noise figure of the opamp is predominantly determined by the first stage. Because of their better noise behavior and higher voltage gain, a P-type input differential pair is used again. As the opamp is a part of a feed-back loop, the open loop gain is not a critical parameter. Therefore the channel length of transistor $M6$ is set to it's minimum value ($L_6 = 2.4\mu m$). This is necessary to achieve the monotonic settling as $M6$'s gate-source capacitance is minimal. In order to achieve maximum input and output common mode ranges, the saturation voltage of the current source transistors

(for instance $\sqrt{\frac{2I_{M7}}{K_p W7/L7}}$) should be as small as possible. Therefore transistors $M5$ and $M7$ are also implemented with minimum channel lengths.

- Clamping transistors and charge dumping transistors. The matching of the clamping transistors (M_{cl1} ; M_{cl2}) is not critical as the clamping voltage (V_{clamp}) is a compressed function (square root) of the transistors dimensions and the function is a worst case function for which either the exact result is non-critical - outside the extrema of the activation function - or sufficient margin is applied - in normal linear use of the integrator. Minimum length sized transistors are used to assure that the clamping diodes are as fast as possible, so that the modulation of the virtual ground node is minimal. The transistors widths should be large enough to keep the output voltage within the limits of V_{clamp} at the largest possible current (NI_{wmax}). The dumping transistor's (M_{s1} ; M_{s2}) width have to be wide enough to de-charge the integration capacitor (C_I) fast enough.

- Sampling circuitry. The sampling circuitry consists of the sampling switches (M_{s3} ; M_{s4}) and the sampling capacitor (C_S). The sampling switches are minimum sized. The sampling capacitor must be large enough to hold the integrator's output voltage for a pulse width reference signal period (500 nsec). The larger the sampling capacitor, the smaller the offset voltage caused by the sampling.

The dimensions of the integrator are given in table 5.2. The dimensions are chosen through hand calculations (appendix G) and HSPICE simulations.

Table 5.2: Dimensions integrator

Integrator			
Transistor	*W or other*	*L*	*unit*
M_1	9.6	4.8	μm
M_2	9.6	4.8	μm
M_3	4.8	4.8	μm
M_4	4.8	4.8	μm
M_5	16	2.4	μm
M_6	96	2.4	μm
M_7	96	2.4	μm
M_{rn}	4.8	4.8	μm
M_{rp}	12	4.8	μm
M_{cl1}	64	2.4	μm
M_{cl2}	64	2.4	μm
M_{s1}	32	2.4	μm
M_{s2}	32	2.4	μm
M_{s3}	2.4	2.4	μm
M_{s4}	2.4	2.4	μm
C_I	2.0	-	pF
C_C	0.4	-	pF
C_S	0.2	-	pF
I_{M5}	10	-	μA
I_{M7}	60	-	μA

5.4 Dimensioning the comparator

The comparator's topology is depicted in figure 5.3.

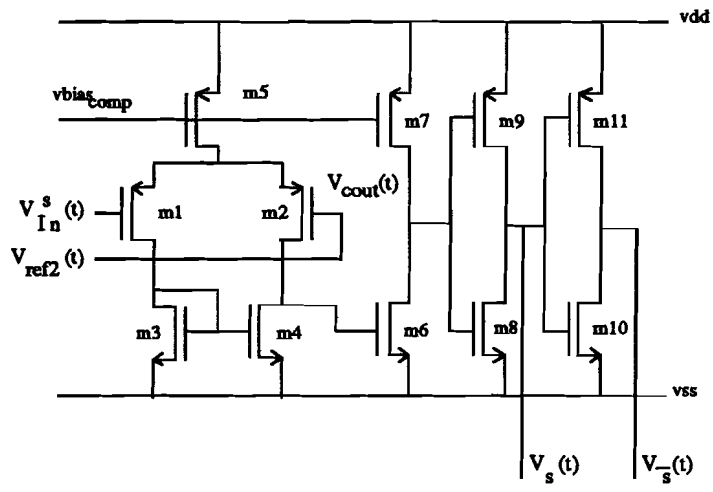

Figure 5.3: The comparator topology.

The following items are taken into account:

- Bias currents. The comparators bias currents ($I_{M5}; I_{M7}$) are determined by the comparator slew rate (SR_{comp}), as derived in appendix F. The rise and fall time of the inverters is not taken into account. This can be done as they are much smaller than those of the comparator.
- DC-operation. The comparator's transistors are assumed to be in saturation. In order to obtain a large common mode input range of $2V_{clamp}$, the (W/L) of input differential pair $(M1; M2)$ and the bias current source $(M5)$ has to be sufficiently large to obtain a small $\|V_{gs} - V_T\|$.
- Noise and mismatch. The noise of the comparator is minimized by using the topology with a P-type input differential pair. Further more, the length of the input transistors has been taken larger than minimal ($L1 = L2 = 4.8\mu m$) to provide sufficient current matching. The offset of the differential pair due to channel length modulation is minimized: The drain voltages of $M1$ and $M2$ are fitted to be equal at the trip point of the comparator.
- The settling behavior. The settling behavior of the comparator is not important as the comparator's output is buffered by inverters.

The dimensions of the comparator are given in table 5.3. The dimensions are chosen through hand calculations (appendix G) and HSPICE simulations.

Table 5.3: Dimensions comparator.

Comparator			
Transistor	W	L	*unit*
M_1	72	4.8	μm
M_2	72	4.8	μm
M_3	9.6	4.8	μm
M_4	9.6	4.8	μm
M_5	96	2.4	μm
M_6	19.2	4.8	μm
M_7	96	2.4	μm
M_8	16	2.4	μm
M_9	48	2.4	μm
M_{10}	16	2.4	μm
M_{11}	48	2.4	μm

Chapter 6

Layouts and simulations of the extracted circuits

The neural hardware is drawn using the DALI [27] layout tool. The extractions of the circuit are made using the Space extraction program [29]. During the layout process, the following rules are taken into account.

- Digital control lines are prevented from crossing sensitive analog circuits.
- Separate digital and analog voltage supplies are used.
- The transistors that require matching are realized as matched structures.

The memory cell's chip area is the most important design issue. Therefore, a maximum number of stacked layers are used for the connections. The memory cells are arranged in a matrix, so the connections have to be transparent to up-down and left-right shifting (abutment). The layout of one memory cell with control logic is given in figure 6.1. The mask colors are, from black to white: *contact or via; metal2; metal1; poly1 and p-bulk*, *n-well* is shaded. The layout of the memory cell measures $176 \times 358 \mu m^2$. The upper part of the cell shows the differential memory cell with its matched pair. The lower part contains the current switches and control logic.

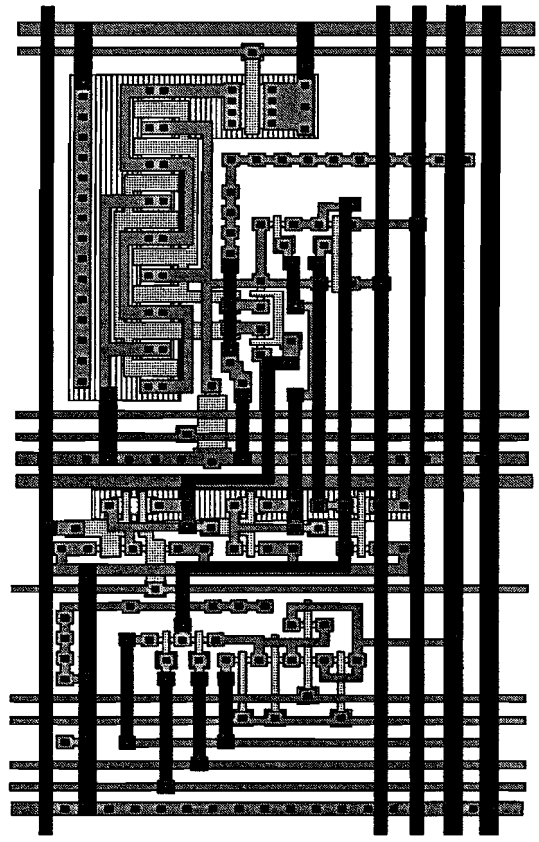

Figure 6.1: Layout of memory cell.

6.0.1 Memory cell: simulation

The DC requirements are addressed by means of measuring the output current as a function of the output voltage. The dominant time constant is measured by applying a current step at the memory cell. The total power consumption of the memory cell is $55\mu W$. The results of the spice simulations are compared to the values calculated by hand (section G.2) given in Table 6.1.

Table 6.1: Constraints differential memory cell.

Differential memory cell					
Formula	*hand*	*spice*	*condition*		*unit*
(G.1)	1.45	0.93	<	1.5	*V*
(G.2)	4.06	2.88	>	2.50	*V*
(G.4)	37	76	<	80	*nsec*

The extracted dominant time constant of the memory cell is larger than the by hand calculated one. This is caused by the fact that the parasitic capacitances are taken into account and the small signal parameters of the simple MOS equations deviate from the more accurate spice models. The simulated common mode output range (CMOR) of the memory cell is $CMOR = 1.95V$. The simulated results meet the constraints.

6.0.2 Discharging of the memory cell

The discharging and charge injection by drain voltage modulation of the memory cell are simulated. The memory cell is initialized, and it's output current is applied to an integrator. The worst case multiplying trajectory of section F.1.1 is used as reference.

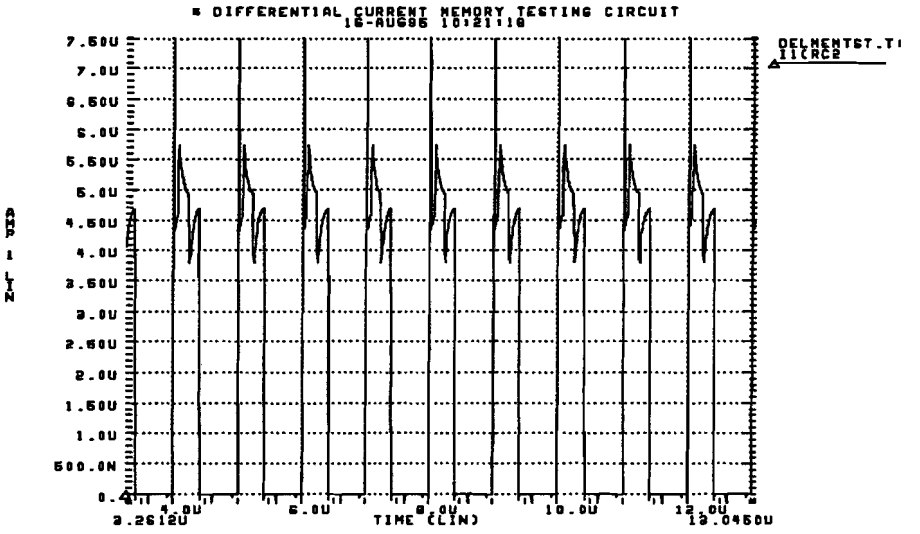

Figure 6.2: Weight current change through gate discharging and drain voltage modulation.

From figure 6.2 it is clear that the modulated drain voltage influences the instantaneous output current. Further more, a permanent change of the memorized current is caused by the modulation. The simulated permanent weight change becomes $\epsilon_{s,t} \approx 0.0072/\,msec$. This value is obtained from a simulation over $0.5\,msec$. This simulation is repeated with tenfold higher accuracy. Because the same results occured, the accuracy of the simulation is assumed sufficient. In order to maintain an 8 bit accurate weight current, the refreshing cycle time has to be $T_r < 2^{-8}/0.0072\,msec = 0.54\,msec$. This value is in accordance with the required refreshing cycle time ($T_r = 0.5\,msec$) in section 5.1.2. The current transfer characteristic of the memory cell is given in figure 6.3.

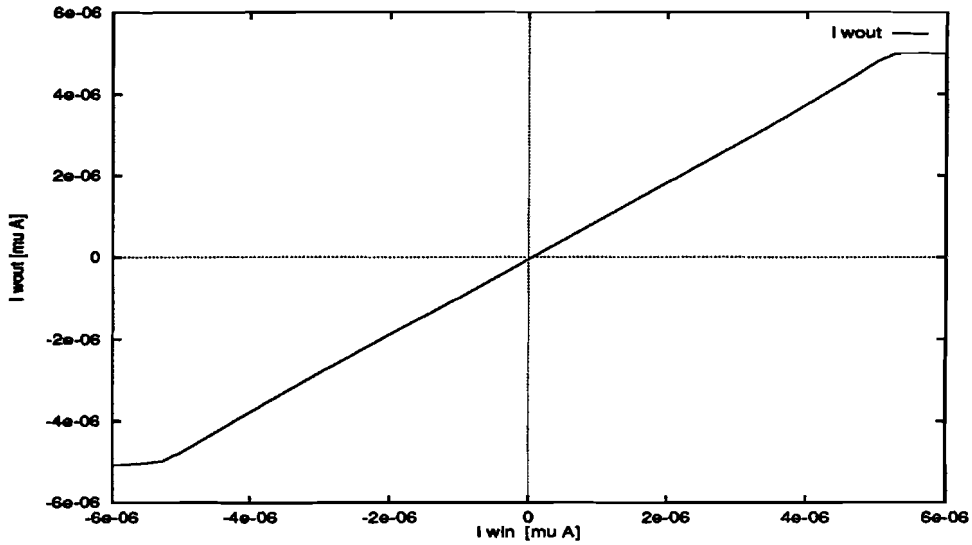

Figure 6.3: Weight current transfer function.

The output current I_{wout} has a small offset ($-50\,nA$). This offset is caused by the non-zero output conductance of the current source $M3$. The charge injection on the gates of $M1$ and $M2$ modulates voltage of the common source node. Due to the non-zero output conductance of transistor $M3$, the output current I_{wout} has a offset. On a system level this offset does not influence proper operation as long as the output current I_{wout} is monotonic with respect to the input weight current (I_{win}).

6.1 Integrator: layout

The integrator's circuit is depicted in figure 5.2. The opamp of the integrator has small sized input transistors. Therefore, no matched structures are used. The mask colors are, from black to white: *contact or via; metal2; metal1; poly1; poly2 and p-bulk, n-well* is shaded. The layout of the integrator is given in figure 6.4. The layout measures $186 \times 309\ \mu m^2$. The integration capacitance C_I can be recognized in the lower part of the layout. The upper part contains the operational amplifier.

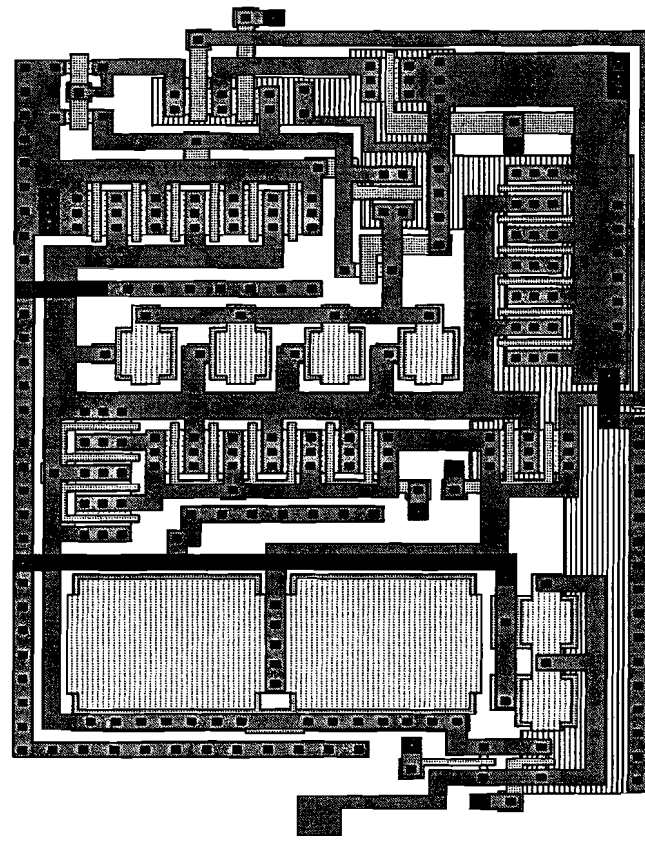

Figure 6.4: Layout of the integrator.

The integrator consists of an operational amplifier and feedback circuitry. The extracted opamp is simulated first.

6.1.1 Opamp: simulation

The DC constraints of the opamp are verified. The opamp's output range is determined in a unity feedback configuration. The opamp's common mode input range is measured with input terminals connected together, and with the output terminal at the reference voltage. The results are given in Table 6.2. The simulated common mode input range equals $CMIR = 3.78\ V$. The simulated output range equals $CMOR = 4.27\ V$. Both values meet the constraints. The total power consumption of the integrator is approximately $0.3\,mW$. Subsequently, an AC analysis is done. The resulting open loop transfer function is given below.

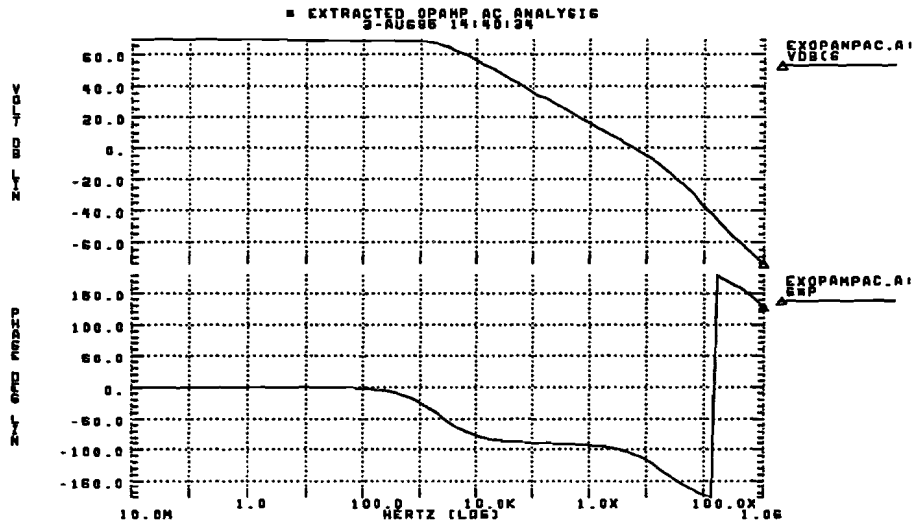

Figure 6.5: Hspice simulation result opamp.

The poles can be determined from this transfer function. The tested constraints are summarized in Table 6.2.

Table 6.2: Constraints integrator: opamp

Opamp						
Formula	*hand*	*spice*	*condition*	*hand*	*spice*	*unit*
(G.5)	2.5		<	2.95	3.19	V
(G.6)	1.5		>	0.65	-0.59	V
(G.8)	0.70		>	0.37	0.45	V
(G.9)	3.30		<	4.58	4.72	V
(G.11)	25	25	>	20		$V/\mu sec$
(G.19)	393	122	>	183	158	$Mrad/sec$

Table 6.2 shows that most of the constraints are fulfilled, except for ??. Instead of a phase margin of 77 degrees, required for monotonic settling the phase margin is 73 degrees. So, some overshoot must be taken into account in the settling behavior.

6.1.2 Integrator: simulation

The clamping voltage of the diodes, and the modulation of the virtual ground node is verified in the following simulation. A maximum integrator input current (NI_{wmax}) pulse is applied to the integrator. The voltage of the virtual ground node (node 3) and sampled output voltage (node 4) is depicted in figure 6.6. The input current is depicted in the lower part.

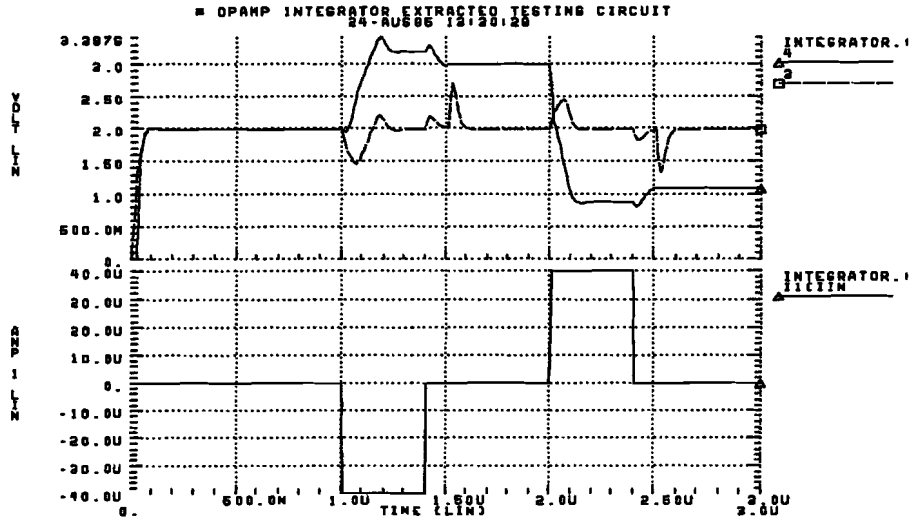

Figure 6.6: Hspice result clamping voltage.

The clamping voltage derived from figure 6.6 equals $V_{clamp-} = 1.14\,V$ for a minimal output voltage and $V_{clamp+} = 1.18\,V$ for maximal output voltage. The difference is caused by the body effect of the clamping transistors. The peak value around $t = 1.2\mu sec$ is caused by the integrator showing large signal behavior. Both values are at the safe side of the assumed clamping voltage of $V_{clamp} = 1.3\,V$.
The simulated voltage modulation on the virtual ground node equals $0.97V$. This meets the constraints ($CMIR = 1\,V$), but the range is not symmetrical around V_{ref} (from 1.46 V to 2.43 V). The peak values around $t = 1.50\,\mu sec$ and $t = 2.50\,\mu sec$ are caused by the dumping of the integrator capacitance (C_I). These peaks do not influence the memory cell because the current switches that connect the memory cells are open in the dumping phase. The simulated common mode input range of the integrator, required for the peak values equals $CMIR = 1.3\,V$. This value lies within the simulated opamp's common mode input range of section 6.1.1: $3.78\,V$.
The non-symmetrical settling behavior (for positive or negative input current) is caused by the non-symmetrical topology of the opamp: For a high output voltage, $M7$ sources the output current, as for a low output voltage, $M6$ dumps the output current. The overshoot around $t = 1.2\,\mu sec$ is caused by the limited sourcing current of transistor $M7$ ($60\,\mu A$). Transistor $M6$ can source a larger current than $M7$, due to a larger transconductance. Therefore, the integrator settles faster for low output voltages. It is obvious that the opamp shows large signal behavior in this case.

6.2 Comparator: layout

The comparator scheme is given in figure 5.3. The layout of the comparator without inverters ($M8 - M10$) is given in figure 6.7. The layout measures $151.6 \times 180\ \mu m^2$. The mask colors are, from black to white: *contact or via; metal2; metal1; poly1 and p-bulk, n-well* is shaded. The bottom-right transistor ($M6$ of figure 5.3) is implemented as a double transistor to provide better matching with transistor $M4$.

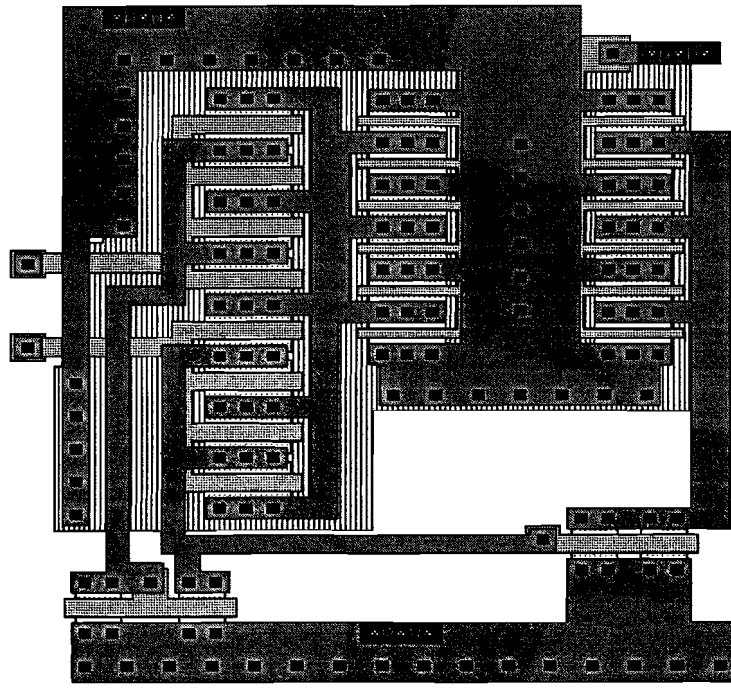

Figure 6.7: Layout of the comparator.

6.2.1 Comparator: simulation

The common mode input range is simulated by making a voltage sweep (period = 30μ sec) on the input terminals. A differential square wave signal (period 100 ns; amplitude = 1 V) is applied at the input. The simulation result is shown in figure 6.8. The node voltages are $V(3) = V_{\bar{s}}(t)$; $V(4) = V_s(t)$ and $V(7)$ shows the input voltage sweep as a function of time.

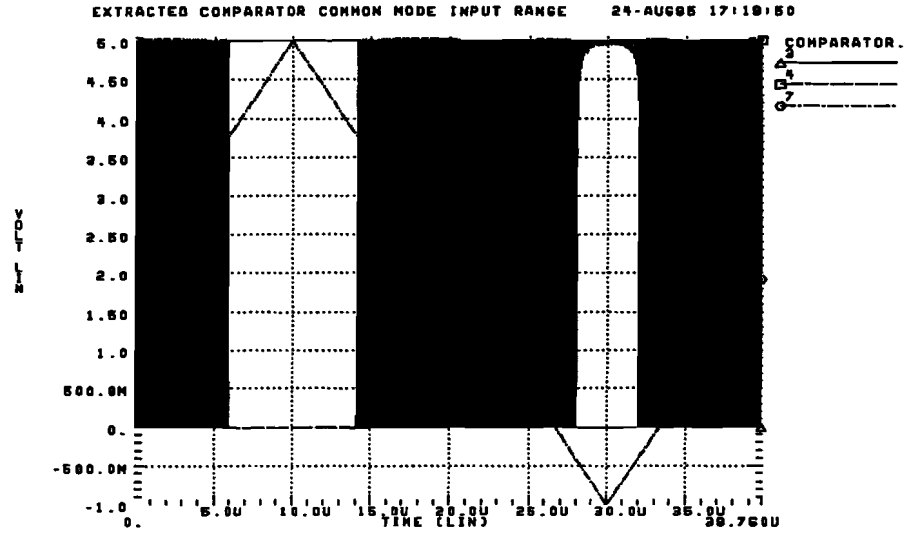

Figure 6.8: Hspice simulation result common mode input range comparator.

The common mode input range, determined from figure 6.8 equals CMIR=4.09V. (from $-0.37V$ to $3.72V$) This result is generated at a bias current of $I_{M5} = I_{M7} = 45\ \mu A$ instead of $I_{M5} = I_{M7} = 15\mu A$ used in appendix G. This is necessary because the by hand calculations did not consider the parasitic capacitances. The extractor includes the parasites that slow down the comparator. It can be speeded up by increasing the bias currents at the expense of common mode input range. The common mode input range determined above meets the constraint of $V_{ref} + / - V_{clamp}$. The total power consumption of the comparator is approximately $0.6mW$. The DC-offset of the comparator can be determined by applying a DC-sweep to the input terminals. The results of the simulation are shown in figure 6.9. In this figure, the nodes are: 7: inverting input; 8: non-inverting input, and 4: the output node of the comparator. The normalized and magnified signals are depicted in the lower half of the figure.

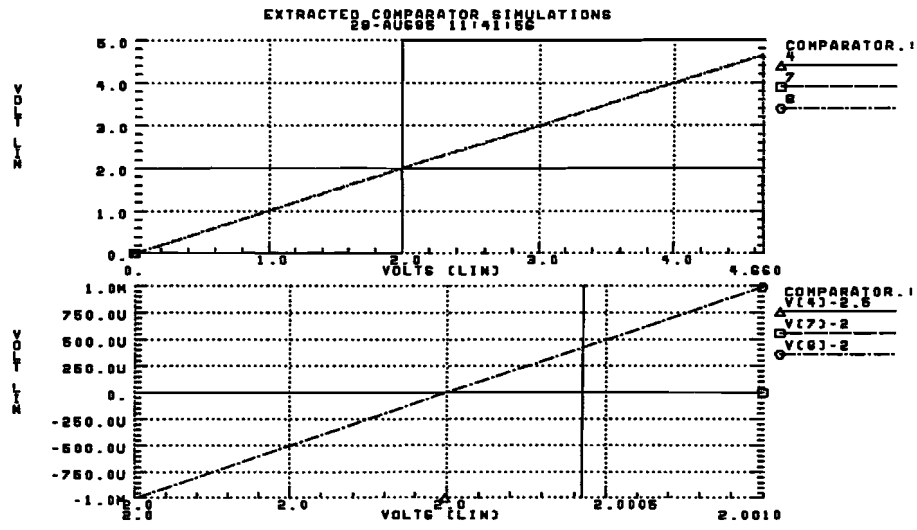

Figure 6.9: Hspice simulation result comparator input offset.

According to this simulation, the offset voltage is smaller than $0.5mV$. This offset is due to channel length modulation. Offset due to mismatch is not taken into account here.

The input voltage ($V_{I_n}^s(t)$) to pulse width (T_{PW}) transfer function and the propagation delay (T_{prop}) of the comparator is determined in the following simulation. The reference voltage has a triangular shape with a peak-to-peak value of $5\,V$, and a period of $500\,nsec$. The results are given in figure 6.10.

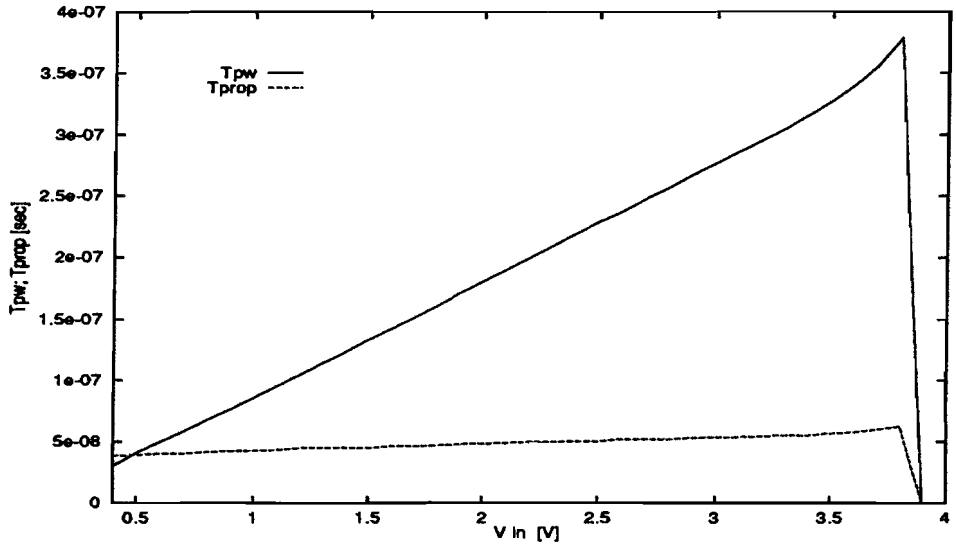

Figure 6.10: Input voltage to pulse width transfer function and propagation delay.

This simulation shows that the input voltage to pulse width transfer function is monotonic

within the common mode range of the comparator. The propagation delay is weakly dependent of the input voltage, and is smaller than 63 *nsec*.

6.3 Synapse: simulation

A synapse, consisting of a memory cell, comparator and integrator is tested. A multiplier transfer function is generated. The input signals are: memory cell input current I_{win} and comparator input voltage $V_{I_n}^s(t)$. The integrator output voltage is plotted in figure 6.11.

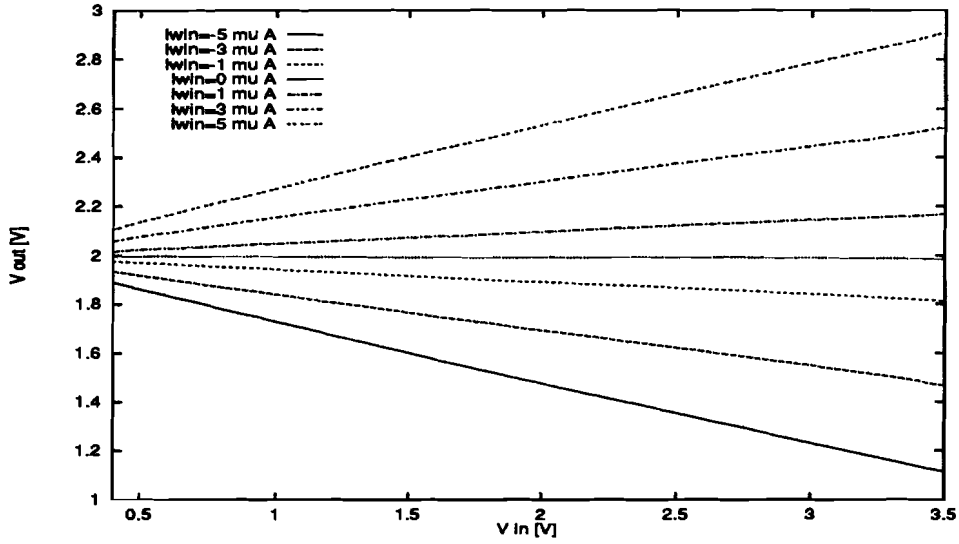

Figure 6.11: Synapse multiplier transfer function.

The absolute error without offsets is depicted in figure 6.12.

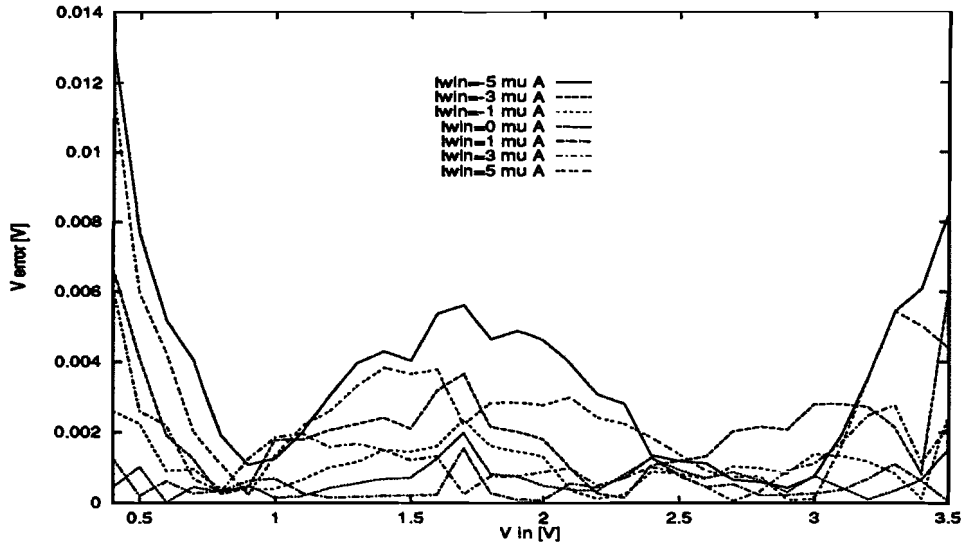

Figure 6.12: Absolute multiplier error.

The random shape of the error voltage is caused by rounding errors of the transient analysis ($ABSVAR = 2e - 2\,V$). As the maximum multiplier output voltage range is ($V_{clamp} \approx 1.1\,V$), the maximum non-linearity of the multiplier (ϵ_m) becomes $\epsilon_m = 1.2$ %.

6.4 Sum of products: simulation

Eight memory cells are connected to one integrator. The worst case reference situation of section F.1.1 is used. In order to prove monotonicity of the sum of products, one of the input signals (V_{I_8}) is varied over its voltage range. A triangular reference voltage with 2 Volt mean value, 1.5 Volt amplitude and 500 nsec. period is used. The required 160 nsec. multiplying period is generated by direct comparison of the triangular reference voltage with a 1.46 Volt input voltage. The results of the simulation are given in figure 6.13. The upper half of the figure gives the sampled output signal (OUT=$V_o^s(t)$). The worst case situation is simulated from $t = 2\,\mu sec$ to $t = 3\,\mu sec$. Subsequently, from $t = 3\,\mu sec$ to $t = 15\,\mu sec$, VIN8=V_{I_8} is varied from $0.9\,V$ to $3.1\,V$.

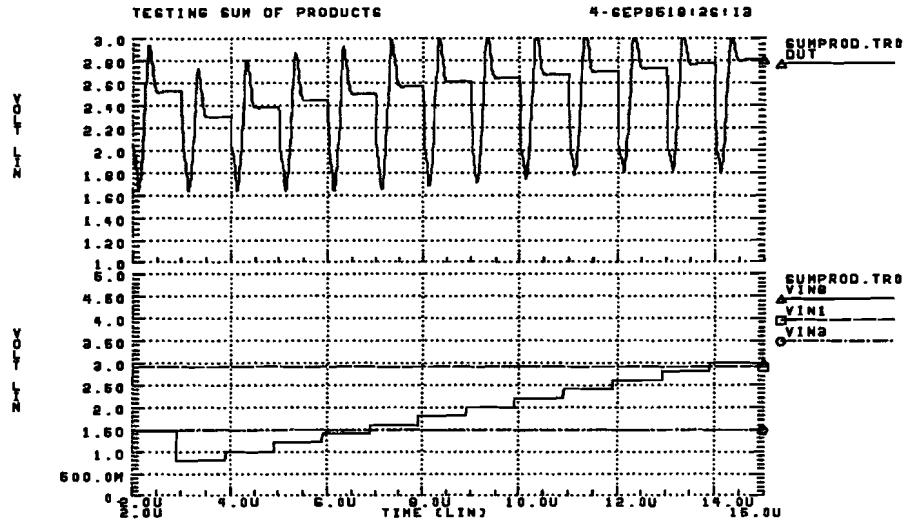

Figure 6.13: Hspice simulation result: sum of products.

The worst case value of 0.5 Volt, is approached whithin 5%. The sum of products result from $t = 3\,\mu sec$ to $t = 15\,\mu sec$ varies in a monotonic manner, although the integrator output voltage is clamped.

6.5 Total chip: layout

The layout of the neural network chip is given in figure 6.14. The chip including the not depicted pad's measures about 3×4 mm^2. The 6×6 synapse matrix can easily be recognized. At the top and the left side, the address selection circuitry can be seen. At the right side a column of 6 comparators is placed. At the bottom, a row of seven integrator and output buffers can be seen. The most left combination is a test pair.

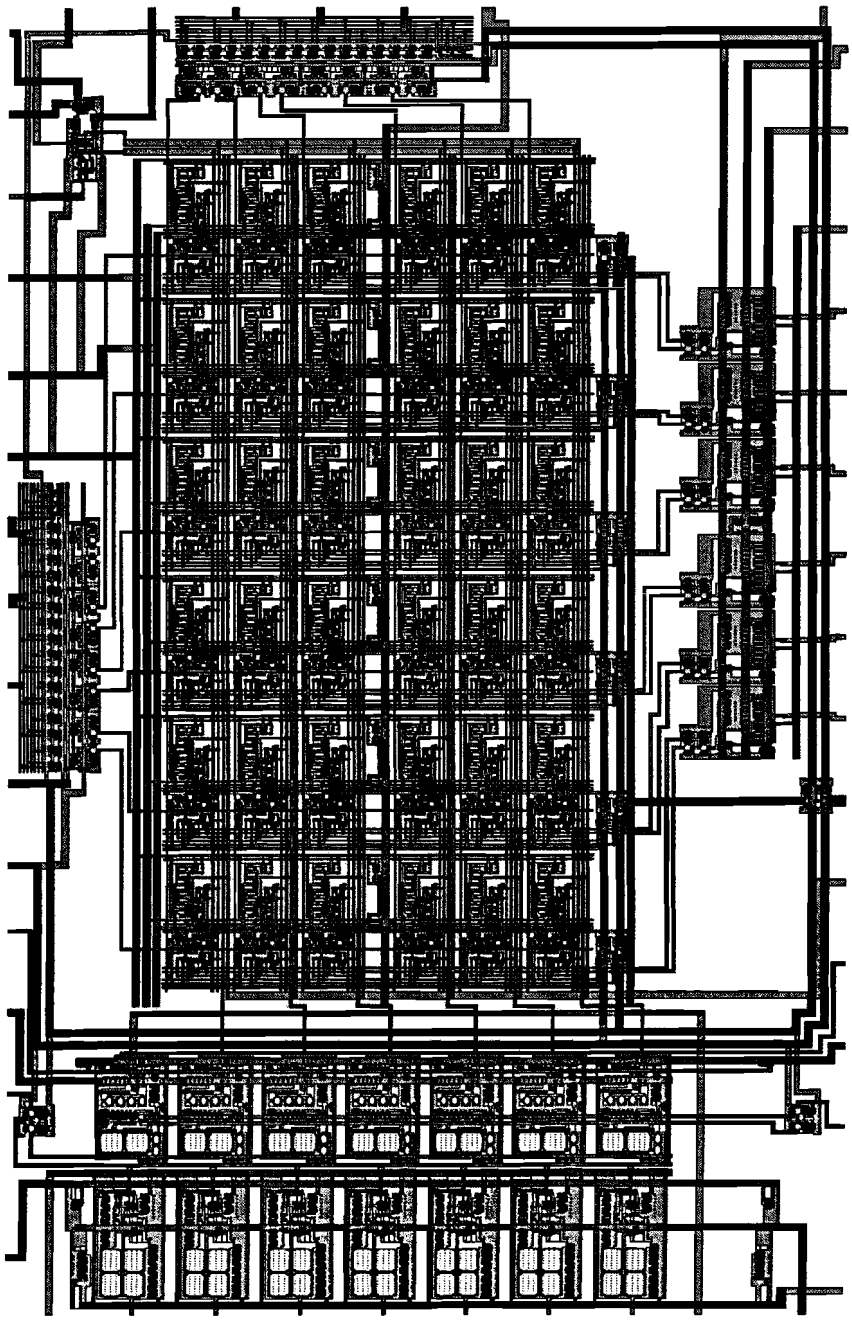

Figure 6.14: Layout of the switched current neural network chip.

6.6 Total chip: simulation

The operation of the chip is checked by initializing three rows of the synapse matrix ($W_{3,n}..W_{5,n}$ with n=1..6) with orthogonal weight vectors, from which one vector contains zero's ($W_{3,n}$=0 with n=1..6). Subsequently, three input vectors ($I_n(1)$, $I_n(2)$, $I_n(3)$ with n=1..6) are applied at 52, 53 and 54 μ sec respectively. The input vectors are chosen in such a way that the sums of products calculations number 3 to 5 are excited subsequently. The simulation is done at low accuracy (ABSMOS=1e-4 RELMOS=1e-3 ABSVAR=1e-1 RELVAR=1e-1) to avoid a long

simulation time. The simulation results are shown in figure 6.15.

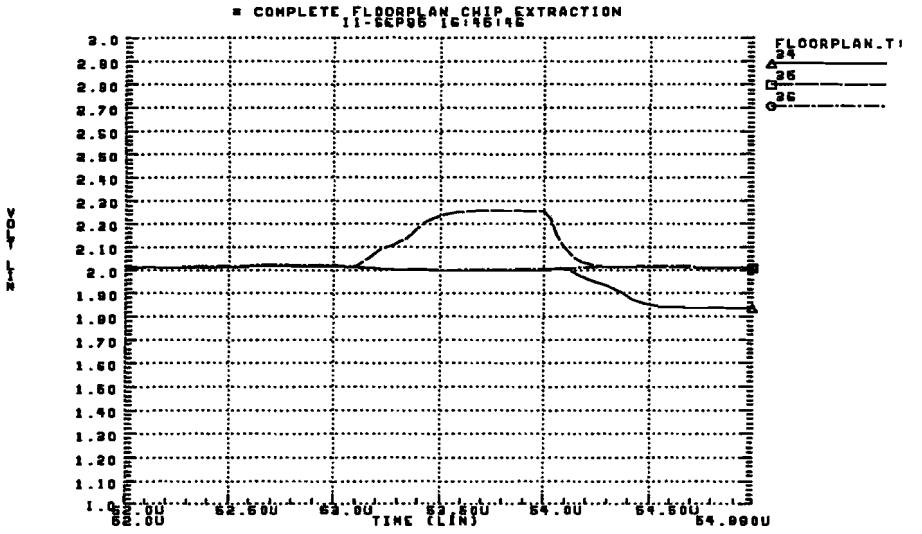

Figure 6.15: Hspice simulation result: Simulation of neural chip.

Signal V(36) shows the zero weight vector sum of products ($\sum_{n=1}^{6} W_{3,n} I_n$ result. Signals V(34) and V(35) show the other orthogonal weight vector results, that are excited subsequently. As the qualitative result is more important than the quantitative result in this simulation, the operation of the chip is shown.

Chapter 7

Conclusions and recommendations

7.1 Conclusions

The designed and implemented switched current neural network will be compared to the reviewed networks of chapter 2. For this comparison, the weight storage and multiplier (synapse) will be addressed together.

7.1.1 The synapse

The most important design issues of the synapse are its power consumption and its area. The design parameters of the switched current synapse are compared to the reviewed synapses in Table 7.1.

Synapses							
Reference	*Quadrants* -	*Area* [λ^2]	*Transistors* -	*Power [Watt]*	(1) %	(2) *bit*	*Speed [MHz]*
[4]	4	1E4	> 30	10E-6	-	7	-
[9]	4	4.4E3	19	-	5	-	4.0
[15]	4	-	4	9E-4	-	-	3.3
[7]	4	5.8E3	4	3E-4	-	-	-
[17]	4	2.0E3	6	4E-4	-	4	-
[10]	4	-	9	-	2-5	-	20
[12]	4	-	5	-	1	-	-
[13]	4	-	-	-	-	8	-
[11]	2	2.1E3	-	3E-3	1	-	1
[16]	4	18.3	0	-	1	-	10
[18, 19]	4	8.6E3	13	2E-4	-	-	10
(*SI synapse*)	2	10.9E3	24	5.5E-5	1.2	7	1

(1): accuracy multiplier; (2): accuracy weight storage.

Table 7.1: Design parameters of synapses

- Chip area: The designed switched current synapse occupies the largest chip area of the table. The design is not competitive to the other reviewed synapses. Some gain can be achieved on this area by: A. Decreasing the memory cell's gate capacitance. B. Using matched structures. C. Using separate digital and analog power supply lines. D. Using

minimum sized supply lines. A lower weight storage accuracy must then be accounted for. Nevertheless, the decrease of chip area will not be dramatical.

- Power consumption: The switched current synapse has, compared to other synapses, a small power consumption. The static power consumption can even be reduced further, by decreasing the bias currents. In order to maintain the weight storage accuracy, the refreshing time (T_{select}) must be increased subsequently.
- Multiplier accuracy: The accuracy of the switched current multiplier using a pulse width modulation technique is comparable to [11] using the same technique.
- Weight storage accuracy: The switched current synapse provides a high weight storage accuracy. The most important reason of permanent weight change is the drain voltage modulation of the current memory. This can be improved in two manners: A. Using cascoded memory cells to make the memory cell less sensitive to drain voltage modulation. B. Using a current-to-current converter at the summing node of the synapses to reduce the modulation. Solution A enlarges the synapse's chip area. Solution B enlarges the neuron's chip area and power consumption.
- Processing speed : The processing speed of the switched current synapse is lower than the speed of analog designs, but comparable to the other [11] pulse width modulation multiplier. The speed of the chip may be increased - using the same technique - by some factors at the expend of multiplier dynamic range and accuracy.

The designed switched current synapse has, compared to analog techniques, a large chip area, a low power consumption, a high accuracy and low processing speed. The relative large chip area of the switched current synapse is obviously the largest disadvantage. It is not expected that a vast reduction of chip area is possible, as the bulk of chip area is used for weight storage and refreshing circuitry.

The power consumption can be decreased quite easily. The synapse's topology is suitable for low voltage applications, so extra power decrease is possible.

The processing speed of the synapse can further be increased by using other multiplication techniques. The pulse width technique achieves a relative high accuracy. The dynamic range is inversely proportional to processing speed. The high accuracy might not be necessary in neural network applications. Alternate techniques such as a current mode or transconductance techniques might give faster multipliers with lower accuracy. The chip area occupation of other multiplier techniques might not be able to compete with only four minimum sized transistor switches.

7.1.2 The neuron

The neuron, consisting of integrator and the comparator, has a total power consumption of about 0.9 mW. The designed neuron - consisting of the summing node and the activation function - is compared to the other activation functions in Table 7.2.

Activation functions				
Reference	*Type*	*Area [lambda ^{2}]*	*Transistors*	*Gain programmable ?*
[4]	channel length mod.	-	25	Y
[9]	Transconductance	4.8E3	13	Y
[15]	Transconductance	-	10	N
[7]	Transconductance	-	4	N
[17]	Transconductance	-	7+opamp	Y
[10]	V/I converter	-	2	N
[12]	Current mode PWL	-	12 or 18	N
[13]	-	-	-	N
[11]	Double inverter	-	4	N
[16]	Double inverter	-	5	N
[18, 19]	A/D converter	-	-	Y
(*SI neuron*)	int./comp.(1)	5.5E3	24	Y

(1): Integrator and comparator.

Table 7.2: Design parameters of activation functions

- Chip area: The designed neurons chip area is comparable to the other known data dealing with micro-chip area. Some chip area can be gained by minimizing transistor lengths. This will worsen the matching properties of the transistors and integrator and comparator input voltage offset will increase.
- Programmability: The designed neuron's gain is programmable by means of changing the shape of the pulse-width modulation reference signal.

As shown above, the switched current neuron is comparable to the reviewed neurons.

7.2 Recommendations

As shown in section 7.1, the most unfavorable property of the switched current synapse is it's chip area. The synapse can be optimized for area by not using a matched structure for the weight memory. Precaution measures, like the separate digital and analog power supply lines, can be omitted to save chip area. Further more, the "AND" operation, implemented by the extra switches ($Ms5$ and $Ms6$) of figure 5.1, can be transfered to the neuron. The "AND" operation, that determines the multiplying phase, can be saved from the synapses chip area and power consumption.

The most simple way of increasing the processing speed of the synapse is to decrease the maximum pulse width of the pulse width modulator, at the expense of power consumption and accuracy. The minimum pulse width can be decreased by using alternative comparators. Other multiplying principles can be considered, but multipliers that require less chip area than the applied multiplier may be hard to find. Moreover, the advantageous combination of activation function and pulse width modulator can not be used.

The power consumption of the synapse can be decreased by decreasing the bias currents. This will influence the refreshing behavior of the weight memory. In this case, redimensioning will be required to keep the memory cell in strong inversion at lower currents.

Bibliography

[1] S.J. Daubert, D. Valancourt, Y Tsividis, "Current copier cells," in *Electronics Letters,* vol.24, pp. 1560-1562, 8th December, 1988.

[2] D. Hammerstrom, "A VLSI Architecture for High Performance, Low Cost, On-Chip Learning." in *Proc. IEEE Int. Joint Conf. on Neural Networks* vol. 2, pp. 537-544, 1990.

[3] Y. Horio, S. Nakamura, "Analog memories for VLSI Neurocomputing," in E. Saukes-Sinencio, ed., *Artificial Neural Networks*, pp. 344-363 , IEEE PRESS, Clifford Lau, 1992.

[4] T. Duong, S.P. Eberhardt, M. Tran, T. Daud, A.P. Thakoor, "Learning and Optimization with Cascaded VLSI Neural Network Building Block Chips," in *Proc. IEEE Int. Joint Conf. Neural Networks*, pp. I-184-189, 1992.

[5] A.J. Monatlvo, P.W. Hollis, J.J. Paulos, "On-Chip Learning in the Analog Domain with Limited Precision Circuits," in *Proc. ISCAS*, pp. 196-201, 1992.

[6] T. Baker, D. Hammerstrom, "Characterisation of Artificial Neural Network Algotithms", in *Proc. ISCAS*, pp. 78-81, 1989.

[7] N.I. Khachab, M. Ismail, "An Analog Continuous-Time Programmable Neural Network", in *Proc. ISCAS*, pp. 282-288, 1990.

[8] J.E. Hansen, J.K. Skelton, D.J. Allstot, "A Time-Multiplexed Switched Capacitor Circuit for Neural Network Applications", in *Proc. ISCAS,* vol. 4, pp. 2177-2180, 1989.

[9] J. Choi, B.J. Sheu, "VLSI Design of Compact and High Precision Neural Network Processors," in *Proc. IEEE Int. Joint Conf. on Neural Networks,* vol. 1, pp. 637-642, 1992.

[10] Y. Cao, S. Mattison, "Current-Mode Analog Neural Network Circuits for High-Speed Applications", in *ECCTD '93 - Circuit Theory and Design,*, Elsevier Science Publishers, pp. 263-268, 1993.

[11] B.J. Maundy, E.I. El-Masry, "A Self-Organizing Switched-Capacitor Neural Network", in *IEEE trans. on Circ. and Syst.*, vol. 38, pp. 1556-1563, 1991.

[12] M. Delgado-Restituto, A. Rodriguez-Vazquez, "Current-Mode Building Blocks for CMOS-VLSI Design of Chaotic Neural Networks", in *Proc IEEE Int. Conf. on Neural Networks* , vol. 3, pp. 1993-1997, 1994.

[13] V. Peiris, B. Hochet, M. Declerq, "Implementation of a fully Parallel Kohonen Map: A Mixed Analog Digital Approach," in *Proc. IEEE Int. Joint Conf. on Neural Networks,* vol. 4, pp. 2064-2096, 1994.

[14] B. Hochet, V. Peiris, S. Abdo, M. Declerq, "Implementation of a Learning Kohonen Neuron," in *IEEE Journal of Sol. State Circ.*, vol. 26, pp. 262-267, 1991.

[15] H. Harrer, J.A. Nossek, R. Stelzl, "An analog Implementation of Discrete-Time Cellular Neural Networks", in *IEEE trans. on Neural Networks,* vol. 3, pp. 466-476, 1992.

[16] U. Cilingiroglu, "A Purely Capacitive Synaptic Matrix for Fixed-Weight Neural Networks", in *IEEE trans. on Circ. and Syst.* vol. 38, pp. 210-217, 1991.

[17] M. Holler, S. Tam, H. Castro, R. Benson, "An Electrically Trainable Artificial Neural Network (ETANN) with 10240 "Floating Gate" Synapses", in *Proc. IEEE Int. Joint Conf. on Neural Networks*, vol. 2, pp. 191-196, 1989.

[18] B.E. Boser, E. Sackinger, J. Bromley, Y. LeCun, R.E. Howard, L.D. Jackel, "An Analog Neural Network Processor and its Application to High-Speed Character Recognition", in *Proc. IEEE Int. Joint Conf. on Neural Networks*, vol. 1, pp. 415-420, 1991.

[19] B.E. Boser, E. Sackinger, "An Analog Neural Network Processor with Programmable Network Topology", in *Proc. IEEE Int. Sol. Stat. Circ. Conf.,* vol. 10, pp. 184-185, 1991.

[20] G.C. Temes, P. Deval, V. Valencic, "SC-Circuits: The State of the Art Compared to SI Techniques," in *Proc. ISCAS*, vol. 2, pp. 1231-1234, 1993.

[21] S. Churcher, D.J. Baxter, A. Hamilton, A.F. Murray, H.M. Reekie, "The EPSILON Chip - An Anlogue VLSI Neural Net Building Block", in *Proc. Micr. for Neural Networks*, vol. 3, pp. 217-225, April 1993.

[22] J.B. Hughes, W. Redman-White, "Switched-Current Limitations and Non-Ideal Behaviour" in C. Toumazou, J.B. Hughes, N.C. Battersby, eds. *Switched-Currents an analogue technique for digital technology*, pp. 71-135, IEE CIRCUITS AND SYSTEMS SERIES 5, Peter Peregrinus Ltd. on behalf of the Institution of Electrical Engineers, London, United Kingdom, 1993.

[23] J.B. Hughes, I.C. Macbeth, D.M. Pattullo, "Second generation switched-current circuits" in *Proc. IEEE Int. Symp. on Circ. and Sys.*, pp. 2805-2808, 1990.

[24] J.B. Hughes, K.W. Moulding, D.M. Pattullo, "Switched-Current Circuit Design Techniques" in C. Toumazou, J.B. Hughes, N.C. Battersby, eds. *Switched-Currents an analogue technique for digital technology*, pp. 156-190, IEE CIRCUITS AND SYSTEMS SERIES 5, Peter Peregrinus Ltd. on behalf of the Institution of Electrical Engineers, London, United Kingdom, 1993.

[25] J. Daubert, D. Vallancourt, Y. P. Tsividis, "Current copier cells," in *Electronics letters,* vol. 24, pp. 1560-1562, December 1988.

[26] P. E. Allen, D.R. Holberg, "CMOS Analog Circuit Design", p. 374-396. Holt, Rinehart and Winston, Inc., New York, 1987.

[27] P. van der Wolf, J. Liedorp, "DALI USER'S MANUAL", Department of Electrical Engineering, Delft University of Technology, The Netherlands, November 1987.

[28] HSPICE H9007B "User's Manual", Meta-Software, Inc., 1300 White Oaks road, Campbell, CA 95008, September 1991.

[29] A.J. van Genderen, N.P. van der Meijs, "SPACE USER'S MANUAL", *Report ET-NT 94.37* , Department of Electrical Engineering, Delft University of Technology, The Netherlands, October 1994.

Appendix A

List of Symbols

Symbol	Description
Signals	
O_j	output signal neuron j
net_j	sum of products neuron j
$W_{j,n}$	Weight from input j to neuron n
$I_n(^s)(^u)$	Signed (^{s}) or unsigned (^{u}) input signal n
lw_{lim}, hw_{lim}	Lower and higher weight limit value
li_{lim}, hi_{lim}	Lower and higher input signal limit value
ϕ_n	Multiplying phase
ϕ_m	Reseting phase
ϕ_{sample}	Sampling phase integrator output signal
ϕ_{dump}	Dumping phase integration capacitor
Voltages	
$V_s(t)$	pulse stream signal
$V_{I_n}^s(t)$	Sampled voltage input signal n
vdd	Supply voltage
vss	Supply voltage
Vref	Reference voltage
$V_{ref4}(t)$	Reference voltage for four quadrant multiplier
$V_{ref2}(t)$	Reference voltage for two quadrant multiplier
$V_o(t)$	Multiplication result
$V_o^s(t)$	Sampled multiplication result
v_{imin}, v_{imax}	Lower and higher input voltage limit value
V_{clamp}	Integrator clamping voltage
$V_{act_{max}}$	Activation function's extreme input value
v_{bias3}	Bias voltage transistor M3 memory cell
v_{bias4}	Bias voltage transistor M4 memory cell
$vbias_{comp}$	Bias voltage comparator
$vbias_{int}$	Bias voltage integrator
Currents	
I_w	Weight current
I_{wmax}	Maximum weight current
J,$2J$	Bias currents weight memory
$I_{win}; I_{wout}$	Memory cell refresh and output current

Timing

T_0	System clock (ϕ_n) high period
T_s	Pulse width reference signal period
T_{select}	Memory cell's refresh time
T_r	Memory cell's refresh period
T_{PWmax}	Maximum multiplying pulse width
T_{PWmin}	Minimum multiplying pulse width
T_{prop}	Propagation delay comparator
T_{PW}	Pulse width of pulse width modulated signal ($V_s(t)$)

System requirements

ϵ_s	Refreshing error
E_{in}	Input weight accuracy
CMOR	Common mode output range
CMIR	Common mode input range

Various

C_w	Wiring capacitance of memory cell input
C_{gs}	Gate-source capacitance of MOS-transistor
C_I	Integration capacitor
C_S	Sampling capacitor
Mx	Transistor number x
τ	Dominant time constant memory cell
$W; L$	Width and length of channel MOS transistor
K_p	K-factor MOS-transistor
SR	Slew rate
A_0	DC gain opamp
p_0	Dominant pole opamp
N	Number of synapses per neuron
ϵ_m	Maximum non-linearity multiplier
I_{Mx}	Drain-source current through transistor number x

Appendix B

SPICE parameters

The Mietec 2.4 μm N-well C-MOS process:

```
.MODEL N30   NMOS(   LEVEL=2      TOX=42.5N     XJ=0.3U
+VTO=0.85    NSUB=1.16E15 DELTA=1.43  LAMBDA=0.046  NFS=2.64E11
+UO=620      UCRIT=5.2E4  UEXP=0.104  LD=0U        WD=0.1U
+RSH=33.36   JS=1E-3      DELL=0U     AF=1          KF=1.0E-28
+CGSO=0.8E-10 CGDO=0.8E-10 PB=0.8     FC=0.5        DW=0U
+CJ=0.9E-4   MJ=0.5       CJSW=3.3E-10 MJSW=0.33)

.MODEL P30   PMOS(   LEVEL=2      TOX=42.5N     XJ=0.05U
+VTO=-0.85   NSUB=9.3E15  DELTA=1.93  LAMBDA=0.027  NFS=1.36E11
+UO=210      UCRIT=9.4E4  UEXP=0.286  LD=0.25U      WD=0.2U
+RSH=350     JS=1E-3      DELL=0U     AF=1          KF=3.0E-30
+CGSO=2.8E-10 CGDO=2.8E-10 PB=0.8     FC=0.5        DW=0U
+CJ=3.2E-4   MJ=0.5       CJSW=4.0E-10 MJSW=0.33)
```

Appendix C

Grading the neural hardware

C.1 Transconductance mode circuits

- Cascaded VLSI Neural Network Building Block Chips are described by [4], and consist of two types of chips. The first type contains a 31 × 32 synapse matrix and 32 neurons, the second type only contains a 32 × 32 synapse matrix. The chips are fabricated using a 2- micron Complementary MOS (CMOS) technology. The multiplier is a 7- bit Multiplying Digital-to-Analog Converter, and takes an area of about 100 × 200 micron, and 30 transistors. The variable gain activation function uses a MOS transistors channel-length modulation to shape the activation function. The activation function takes 25 transistors. Weights are stored on chip using digital latch memory devices. A 7 bit latch memory takes about 100 × 200 micron. Two types of learning algorithms have been implemented in a chip-in-loop configuration (off chip learning). Cascade Backpropagation requires weight accuracy of at least 10 bits. This accuracy is obtained (12- 13 bits) by paralleling a synapse chip with adapted synapse gain to the synapse part of the neuron chip. Also an 7 × 7 assignment problem was implemented on 4 chips. The 49 neurons with 2401 synapses (fully connected) were trained to solve an optimization problem in a winner-take-all manner. The synapses require voltage supplies of 0 and 10 Volt and one synapse dissipates 10E-6 Watt maximum.

- The VLSI design of compact and high precision analog neural network processors, built in a 2 micron CMOS process is presented in [9]. An improved Gilbert multiplier is used as synapse. It takes $110 times 160$ micron area, and 19 transistors. The accuracy of the synapse is 5%. The synapses multiplying result is summed in a V-I converter and applied to an activation function with a programmable gain. The activation function takes an area of 120 × 160 micron and 13 transistors, the V-I converter 120 × 60 micron. Weights are stored 8 bits accurate as charges on the gates of the multiplier input transistors. Backpropagation learning and competitive learning is implemented in an off-chip learning algorithm. The entire chip using 400 synapses operated at a speed of 4 MHz. The supply voltage is +/- 5 Volt, power consumption is not known.

- A 16 neuron cellular neural network in a 1.5 micron technology is described in [15]. A transconductance multiplier is used consisting of 4 transistors. This multiplier only has to multiply with +/-1, and input range is chosen so that distortion may be neglected. The activation function is realized with an Operational Transconductance Amplifier (OTA) used as a comparator. It consists of 10 transistors, and occupies an area of 90 × 50 micron. Analog weight storage is used as a capacitor voltage. The weight update circuit is implemented off-chip. The capacitor voltage is converted to a current by an OTA, it takes 7 transistors, and 60 × 50 micron area. No learning rule is realized on chip. The

chip operates at a maximum clock rate of 3.3 Mhz. Using a +/- 5 Volt supply voltage, one neuron uses 9E-4 Watt. One cellular neuron occupies an area of 290 × 275 micron.

- A Hopfield type neural net is fabricated [7] in a 2 micron CMOS process. A multiplier-adder is designed, using an operational amplifier and 2(n+1) transistors for n weights. A 16 input multiplier-adder takes an area of about 0.18 mm^2. The activation function consists of a double inverter (4 transistors). Its gain can not be programmed. Weights storage and learning algorithm are implemented off-chip. The chip uses a voltage supply of 0 and 5 Volt, and 5E-3 Watt for 16 synapses and a neuron.

- A commercial available analog neural network using "floating gate" non-volatile analog memory [17] is realized in a special 1.0 micron CMOS process. This special technology supports the floating gates structures. The chip can implement a Hopfield and a one or two layer perceptron neural network. The synapses are modified Gilbert multipliers. One multiplier takes 6 transistors including the two "floating gate" transistors and occupies an area of 41.6x48.3 micron. It dissipates about 2E-4 Watt. No data concerning the accuracy is available. The activation function is implemented as a modified differential amplifier. Its gain can be programmed. It consists of 7 transistors and an operational amplifier. Weights are stored as charges on the "floating gates". The charges are programmable and non-volatile, so no refresh circuitry is needed. An approximate accuracy of 4 bits (6-7%) is obtained. No learning algorithm is implemented on chip. The supply voltage of the chip is +/- 5 Volt.

C.2 Current mode circuits

- A neural network with fixed weights is presented in [10]. The multiplier is implemented as a transconductance amplifier (voltage to current) with a constant transconductance. The multiplier takes 9 transistors, and the circuits area depends on the weight and required accuracy. The activation function is a fixed current to voltage converter consisting of only 2 transistors. Weights are a ratio of transistor widths. Weight accuracy of 2-5 % is achieved. No learning scheme is possible as the network has constant weights. The 20 synapses and 5 neurons chip is fabricated using a 1 micron technology. The supply voltage is 5 Volt, the processing bandwidth is 20 MHz.

- Chaotic neurons with a Nagumo-Sato model and an Aihara model activation functions are implemented in a 1.6 micron CMOS technology [12]. Multiplication is achieved at 1% accuracy by means of tunable transconductance amplifiers using 5 transistors each. The activation functions are realized with 0.2% accuracy using a piecewise-linear CMOS circuit strategy in the current domain. The Nagumo-Sato model takes 12 transistors, the Aihara model takes 18 transistors. Neither weight storage nor learning rules are implemented on chip. The neurons operate at a 3 volt Voltage supply and occupy 0.096 mm^2 for the Nagumo-Sato model and 0.255 mm^2 for the Aihara model respectively.

C.3 Mixed analog digital circuits

- A modular cascadable Kohonen Neuron chip is realized in a standard digital 1.6 micron technology [13]. Each chip contains 4x4 neurons, and each neuron has two synapses. Multiplication is realized by means of pulse width modulation. This network uses a linear activation function, thus only summing of the multiplication results is required. This is done using a capacitor as integrator. Weights are stored in an analog manner as a capacitor voltage [14]. This way, 8 bit accurate weights storage is achieved. The Kohonen learning

rule is implemented on chip. The chip is able to learn at a speed of 800 input vectors per second. The processing speed is 2000 vectors per second. The size of one neuron is 720×880 micron, voltage supply of 5 Volt is required.

- In [11] a Kohonen self organizing network is realized using a 1.2 micron technology. The multiplier is using synchrone pulse arithmetic. A continuous input signal can only be multiplied with zero or one. A synapse takes an area of about 50×60 micron. A sum of products accuracy of 1% is achieved. The activation function is a double inverter hard limiter. Weights are stored as integrator states. The Kohonen learning rule is implemented on chip including a winner take all circuit. A neuron and 5 synapses occupies approximately 60×500 micron. The chip operates at a clock frequency of 1 Mhz and a +/- 6 Volt supply voltage.

- A capacitive synaptic matrix neural network is presented in [16] in a 3 micron CMOS technology. An error correcting neural classifier is implemented. The network corrects 3 bit errors in a 16 input binary pattern. No multipliers are realized on chip. The activation function is realized as a double inverter switched comparator taking 5 transistors. The fixed binary weights are stored as approximately 1% accurate capacitor ratios. No learning rule is implemented on chip. One synapse representing a binary valued weight (+1 or -1) occupies 16.5×10 micron area. The 16 neuron network with 16 synapses for every neuron takes a total area of 662×468 micron. The chip uses a 5 Volt supply voltage, and operates at a speed of 10 MHz.

- A neural network using digital interfacing and analog calculation techniques is fabricated in a 0.9 micron technology [18, 19]. The network can be configured for time delay, feature extraction, fully connected and feedback topologies. The multiplier is a multiplying D/A converter, multiplying the 3 bit input signal with a 6 bit accurate analog weight. It consists of 13 transistors, and occupies an area of approximately $7E-3mm^2$. The maximal power consumption of one multiplier is 2E-4 Watt. The activation functions gain is programmable from 1/8 to 1 in eight steps. Its output is converted to the digital domain by a 3 bit successive approximating A/D converter. Weights are stored as charges on capacitors. Weights are refreshed 6 bit accurate from external digital memory every 110μ *sec*. No learning is implemented on chip. The chip uses a voltage supply of 5 Volt, and operates at a clock speed of 10 MHz.

Appendix D

System design verification

D.1 In detail description of operation of the two quadrant multiplier

The two quadrant multiplier (figure 3.4) is based on pulse width modulation. It operates in two phases. At the beginning of phase $\phi(n)$ at time t_n, the integration device has been reset. The weight current I_w is fed to the integration device during a period T_{PW} in phase $\phi(n)$. This period is generated by immediate comparison of $V^s_{I_{j,n}}(t)$ and $V_{ref2}(t)$. The reference voltage is described in formula D.1.

$$V_{ref2}(t) = \begin{cases} vmax + \frac{2((t-t_{ref0}) mod T_s)}{T_s}(vmin - vmax) & \text{if } t_{ref0} \leq \text{t mod } T_s < t_{ref0} + \frac{T_s}{2} \\ vmin + \frac{2((t-t_{ref0}) mod T_s)}{T_s}(vmax - vmin) & \text{if } t_{ref0} + \frac{T_s}{2} \leq \text{t mod } T_s < t_{ref0} + T_s \end{cases} \quad \text{(D.1)}$$

The input voltage ($V^s_{I_{j,n}}(t)$) has a constant value, because it is held in phase $\phi(n)$. It is compared to $V_{ref2}(t)$ to obtain the pulse width modulated switching signal $V_s(t)$ as shown in formula D.2:

$$V_s(t) = SIGN[V^s_{I_{j,n}}(t) - V_{ref2}(t)] \quad \text{(D.2)}$$

By setting an input signal range, a minimum and maximum pulse width can be set. The pulse width period T_{PW} is the time between the points t_{refpwb} and t_{refpwe} . At these time instances, the reference voltage $V_{ref2}(t)$ is equal to the input voltage $V^s_{I_{j,n}}(t)$. The pulse width period T_{PW} is given by

$$T_{PW} = T_s(\frac{V^s_{I_{j,n}}(t) - vmin}{vmax - vmin}) \quad \text{(D.3)}$$

When an input signal range is set between$vimax$ and$vimin$, the respectively minimum T_{PWmin} and maximum T_{PWmax} pulse width is given in formula D.4:

$$\begin{aligned} T_{PWmin} &= T_s(\tfrac{vimin-vmin}{vmax-vmin}) \\ T_{PWmax} &= T_s(\tfrac{vimax-vmin}{vmax-vmin}) \end{aligned} \quad \text{(D.4)}$$

The two quadrant multiplying switching times are illustrated in figure D.1.

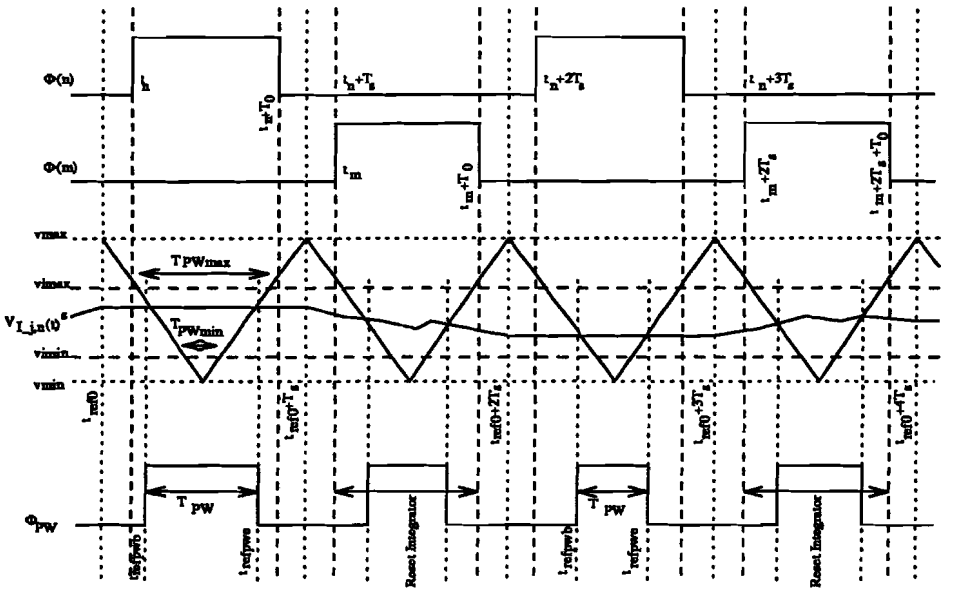

Figure D.1: Two quadrant multiplier switching times

The integration result is reached at time instance $t_n + T_0$. In phase $\phi(m)$, the integrator is reset, and the weight current is prevented from sourcing the reset integrator.

$$V_o(t_n + T_0) = \frac{I_w}{C_I}(T_{PW}) \tag{D.5}$$

D.2 Two quadrant multiplier principle testing scheme

A Hspice simulation has been carried out on the multiplying principle circuit of figure D.2

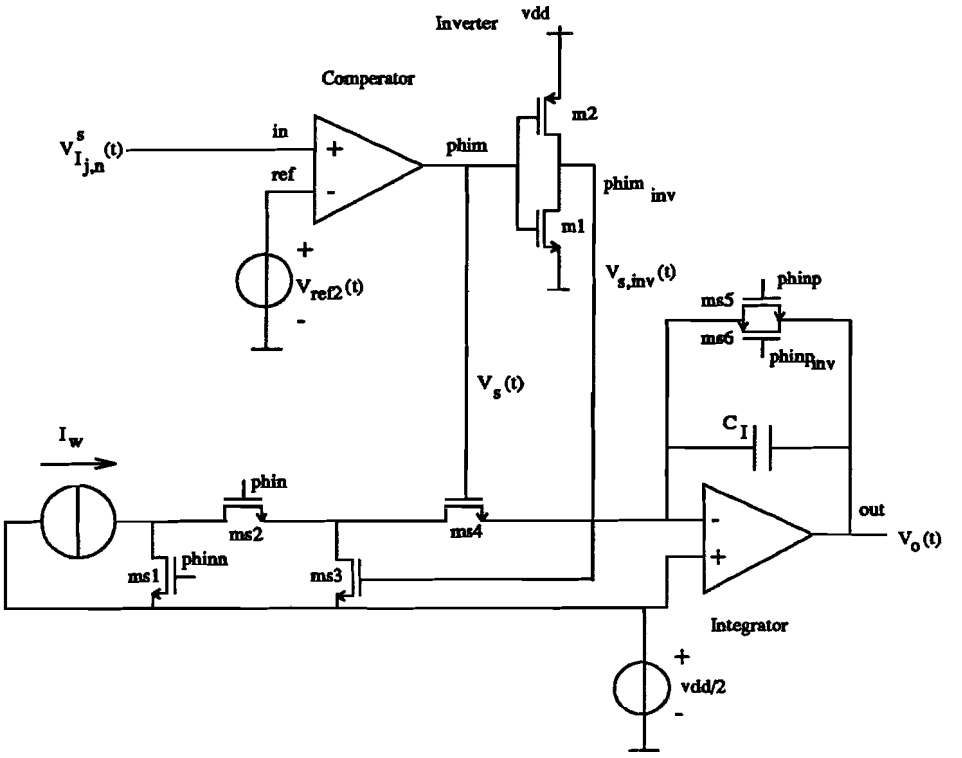

Figure D.2: Hspice two quadrant multiplier principle verification circuit

The operation of the input pulse stream generator and the resulting current through switch $ms4$ is shown in figure D.3. The node voltages are: $V(IN) = V^s_{I_{j,n}}(t); V(REF) = V_{ref2}(t); V(PHIM) = V_s(t)$

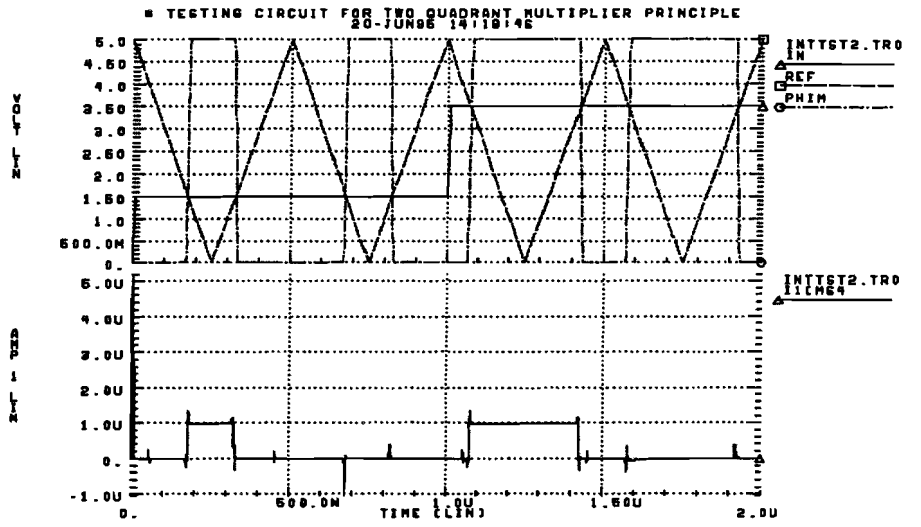

Figure D.3: Hspice simulation results: Input pulse width modulated signal and current through switch $ms4$.

From figure D.3, it is clear that the current through switch $ms4$ is the pulse modulated weight current (I_w). The operation of the integrator is shown in figure D.4.

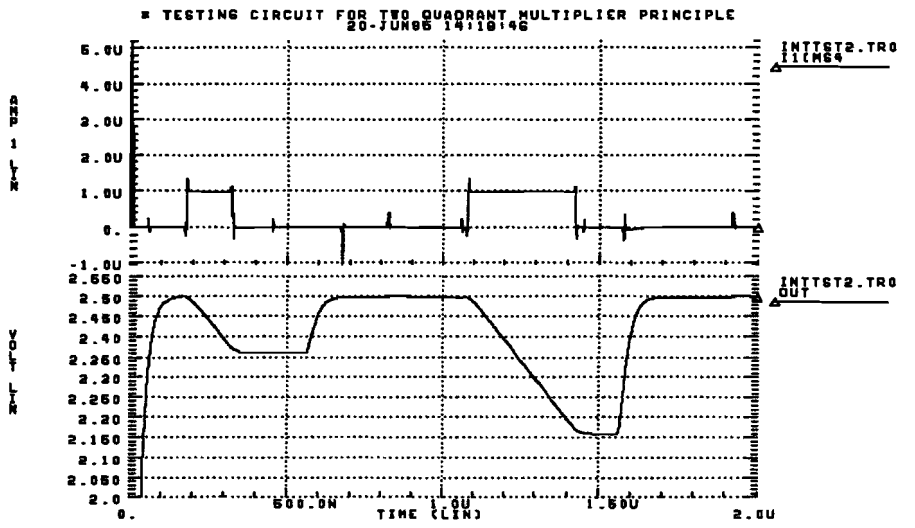

Figure D.4: Hspice simulation results: Integration device output signal and output pulse width generator.

From the simulation results of figure D.4. it is obvious that the integration multiplying result is not ready at the end of phase ϕ_n, due to the limited bandwidth of the integrator's opamp. A certain delay ($T_{s_{int}}$) has to be taken into account for the integrator to be settled. This settling time is derived in section F.1.4. The result: $V_o(t_n + T_o + T_d)$ is proportional to the pulse width (T_{PW}) and the weight (I_w).

Appendix E

Weight memory error sources

The most important weight memory error sources are: Charge injection, settling time, impedance ratios and noise. The circuit of figure 4.3 is used for analysis. The analysis is taken from [22].

E.1 Charge injection

There are two effects of charge injection that influence the stored weight current:

- Charge injection through the sampling switches.
- Charge injection through drain voltage modulation.

These effects will be addressed in the subsequent sections.

E.1.1 Charge injection by the sampling switches

The approximate analysis of the charge injection of a N-MOS differential current memory (figure E.1) is given here. Weak inversion effects are not taken into account.

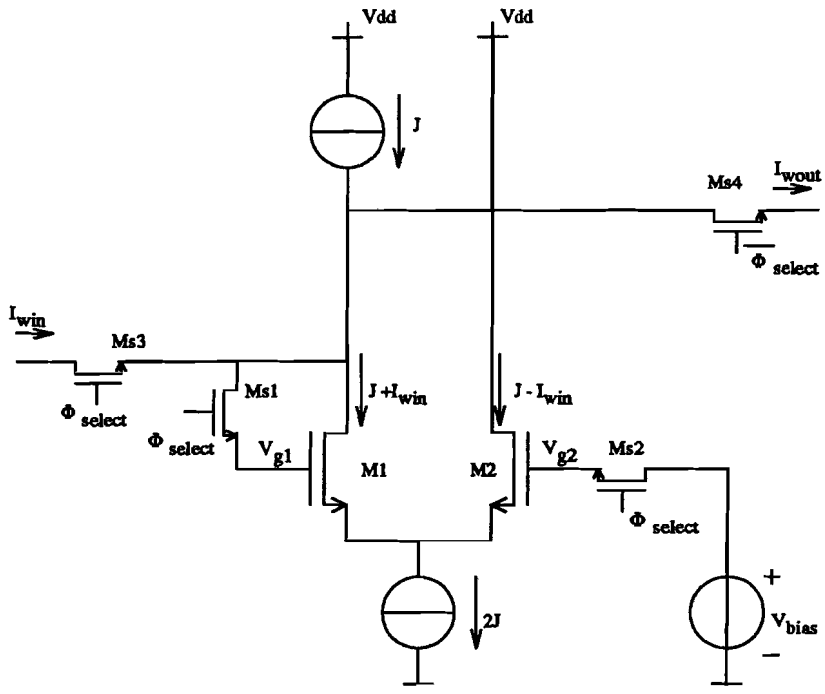

Figure E.1: N-MOS differential current memory

The charge injection of the switches occurs during the opening of the switches. The analysis starts at the time both switches are closed, so the select voltage $V(\phi_{select}) = V_H$. The transistors $M1$ and $M2$ as well as $Ms1$ and $Ms2$ are considered equal. If the input weight current is positive ($I_{win} > 0$), the voltage on the gate of $M1$ (V_{g1}) is larger than V_{g2}. From time t_0, the select voltage is considered to decrease at a constant slope. Switch $Ms1$ is switched off when $V(\phi_{select}) = V_{ck1}$, $Ms2$ when $V(\phi_{select}) = V_{ck2}$. Switch $Ms1$ is switched off earlier than $Ms2$ because of the different gate voltages of transistors $M1$ and $M2$. This is depicted in figure E.2.

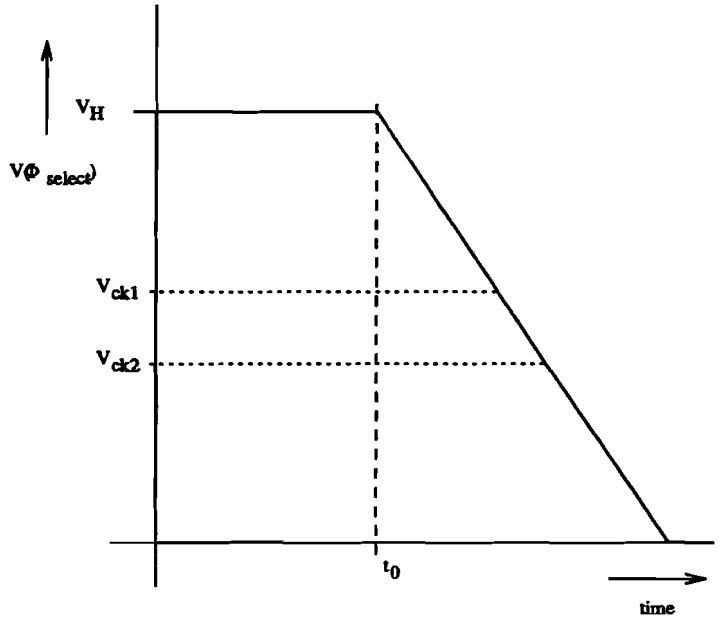

Figure E.2: Switching voltage.

The threshold voltage of the N-MOS switches is approximated as shown in formula E.1.

$$\begin{aligned} V_{TMs1} &= V_{Tn0} + \gamma(\sqrt{|2\phi_n| + V_{g1}} - \sqrt{|2\phi_n|}) \\ \Rightarrow V_{TMs1} &\approx V_{Tn0} + \tfrac{\gamma}{3} V_{g1} \end{aligned} \tag{E.1}$$

The last equation is a rule-of-the-thumb approximation for the body effect. The units of this equation do not match. When the transistor parameters of chapter 5 are used, the approximation is useful in the range of source-bulk voltages from ($0V \leq V_{sb} < 3V$). This is illustrated in figure E.3.

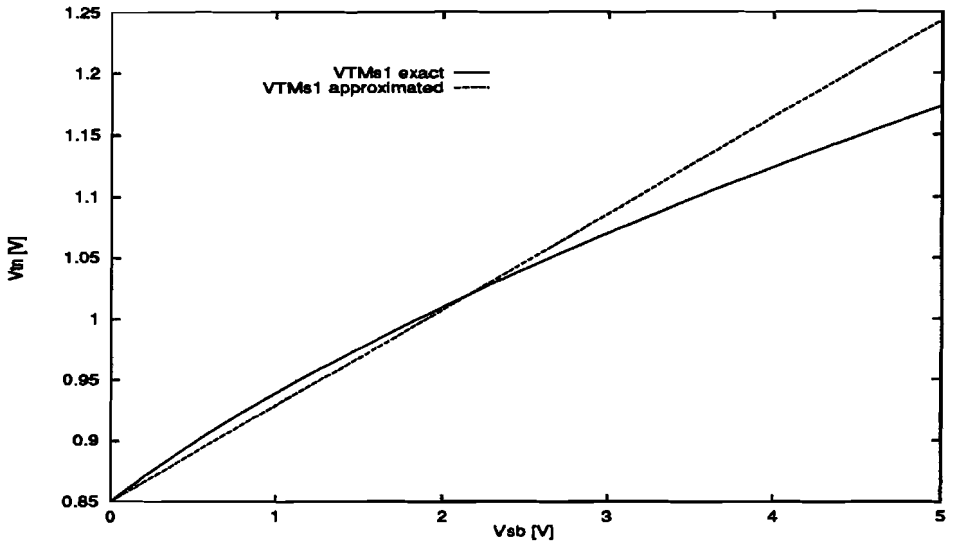

Figure E.3: Approximation of threshold voltage of Ms1 and Ms2 .

The channel charge of the switching transistor $Ms1$ and $Ms2$ before switching is given by:

$$\begin{aligned} Q_{chMs1} &= C_{ch}(V_H - (1+\tfrac{\gamma}{3})V_{g1} - V_{Tn0}) \\ Q_{chMs2} &= C_{ch}(V_H - (1+\tfrac{\gamma}{3})V_{g2} - V_{Tn0}) \end{aligned} \tag{E.2}$$

The channel capacitance C_{ch} can be approximated as $C_{ch} \approx W_{Ms}L_{Ms}C_{ox\square}$. The switching transistors width and length are W_{Ms} and L_{Ms} respectively. The specific gate capacitance is denoted as $C_{ox\square}$. The differential charge q_{dif} injected on the gates of the memory transistors $M1$ and $M2$ depends on their gate voltages:

$$q_{dif} = \alpha C_{ch}(1+\frac{\gamma}{3})(V_{g1} - V_{g2}) \tag{E.3}$$

The factor α is the portion of channel charge that is injected. This portion is determined by the charge distribution of the switching transistor's channel ($0.5 < \alpha < 1$). The resulting difference error voltage δV_g is given in formula E.4.

$$\delta V_g = \alpha\frac{C_{ch}}{C_{gs}}(1+\frac{\gamma}{3})(V_{g1} - V_{g2}) \tag{E.4}$$

The capacitance C_{gs} is the gate source capacitance of the memory transistors $M1$ and $M2$. Formula E.4 is a function of gate voltages, it can be rewritten to a function of gate-source voltages (E.5) as the sources of the memory transistors are connected.

$$\delta V_g = \alpha\frac{C_{ch}}{C_{gs}}(1+\frac{\gamma}{3})(V_{gs1} - V_{gs2}) \tag{E.5}$$

The difference error voltage δV_g causes a difference error current δi. A small signal approximation (E.6) can be used to calculate δi.

$$\delta i = gm_D \delta V_g \tag{E.6}$$

The differential transconductance gm_D is given by $gm_D = \frac{gm1gm2}{gm1+gm2}$. The difference error current must be written as a function of the input weight current I_{win}. The transconductances can be written as a function of the transistors currents (E.7).

$$\begin{aligned} gm1 &= \sqrt{2\beta(J+I_{win})} \approx \sqrt{2\beta J}(1+\tfrac{I_{win}}{2J}) \\ gm2 &= \sqrt{2\beta(J-I_{win})} \approx \sqrt{2\beta J}(1-\tfrac{I_{win}}{2J}) \end{aligned} \tag{E.7}$$

The approximations are valid for $I_{win} \ll J$. Formula E.5 can be rewritten to a function of currents:

$$\delta V_g = 2\alpha\frac{C_{ch}}{C_{gs}}(1+\frac{\gamma}{3})(\frac{(J+I_{win})}{gm1} - \frac{(J-I_{win})}{gm2}) \tag{E.8}$$

This can be rewritten by means of substitution into formula E.6.

$$\delta i = 2\alpha\frac{C_{ch}}{C_{gs}}(1+\frac{\gamma}{3})(I_{win} - J\frac{gm1-gm2}{gm1+gm2}) \tag{E.9}$$

When the approximations of formula E.7 are used, (E.9) can be rewritten as:

$$\delta i = \alpha\frac{C_{ch}}{C_{gs}}(1+\frac{\gamma}{3})I_{win} \tag{E.10}$$

From this formula, it is clear that the difference error current δi is proportional to I_{win}, so despite of the injection error, the weight current remains monotonic with the weight value.

E.1.2 Charge injection by drain voltage modulation

During the holding phase, (ϕ_{select}) is low. The output memory transistor ($M1$) is connected to the virtual ground of the integrator. Due to the limited bandwidth of the opamp, the voltage of the virtual ground node ($V_{in,int}(t)$) will vary. The situation occurs as depicted in figure E.4.

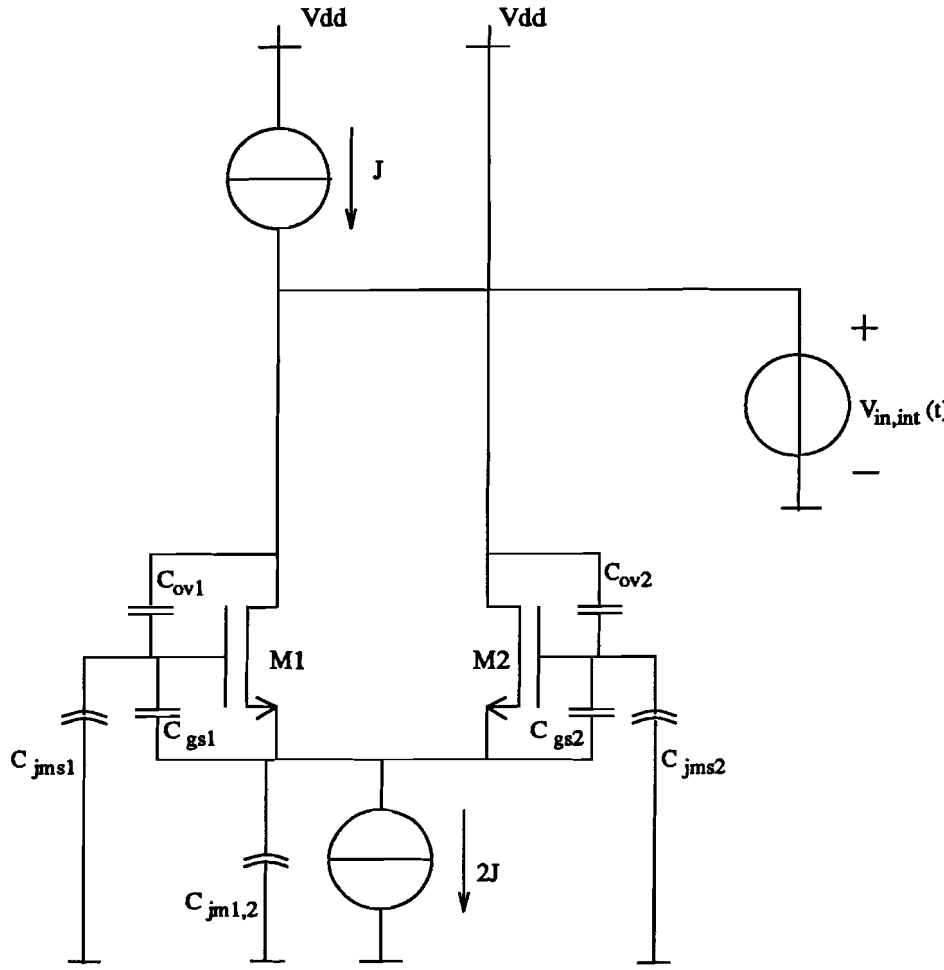

Figure E.4: Weight change through drain voltage modulation.

The junction capacitances ($C_{jms1}; C_{jms2}; C_{jm1,2}$) are non-linear. When the memory transistors are in saturation, the gate-source capacitances ($C_{gs1}; C_{gs2}$) are weakly non-linear. The overlap capacitances ($C_{ov1}; C_{ov2}$) are linear. Due to the non-linearities, a modulation of the drain of transistor $M1$ results in a permanent change of the charge on the memory transistor's gate-source capacitances. This change affects the weight value. The effects of the drain modulation on the memorized weight will be quantified by means of simulation (section 6.0.2). The drain voltage modulation depends on the switching instances of the current switches and the weight current values of all the synapses applied to the integrator. A rational estimate of the drain modulating signal is therefore hard to give.

E.2 Settling behaviour

The settling time error of the weight memory is caused by the time constants of the memory cell. The settling error (ϵ_s) is maximal when two weight memories with opposite signed weights ($I_{win1} = I_{wmax}; I_{win2} = -I_{wmax}$) are initialized subsequently. The simplified small signal diagram of these two memory cells is depicted in figure E.5.

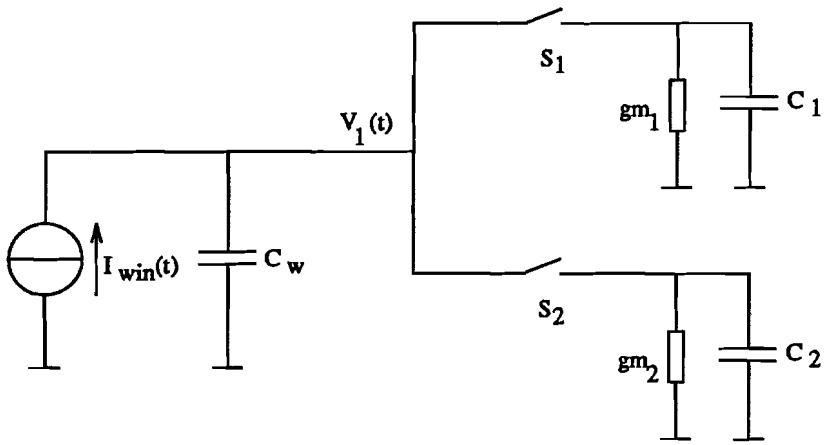

Figure E.5: Scheme for settling error analysis.

The following assumptions are made: The system is in a zero initial state, the settling behavior is linear, and the input current $I_{win}(t)$ can be switched fast enough. The timing diagram of the switches and the signals are given in figure E.6.

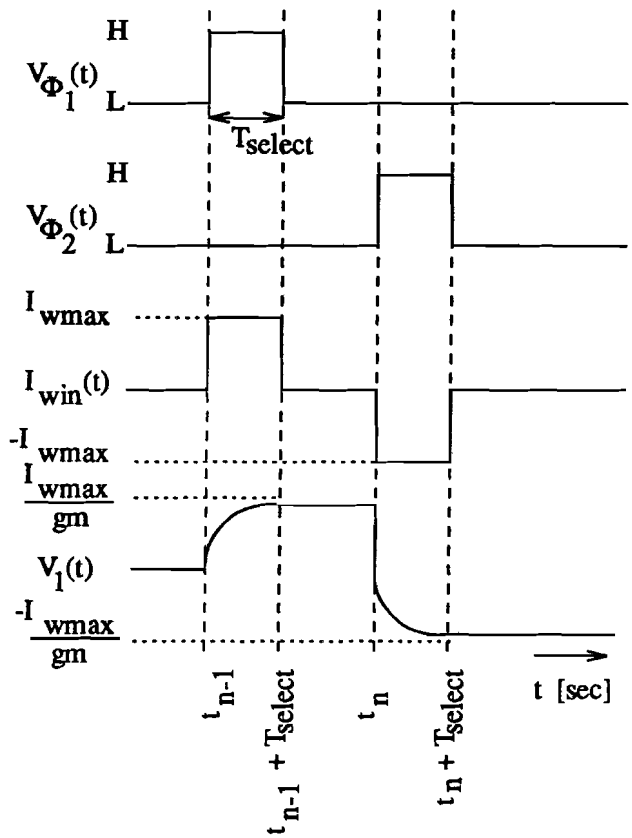

Figure E.6: Timing diagram for settling error analysis.

The transconductances and capacitances of the memory cells are assumed to be equal ($gm_1 = gm_2 = gm; C_1 = C_2 = C$). When $V_{\phi_1}(t)$ is high, switch S_1 is closed and memory number 1 is initialized. At time $V_1(t_{n-1} + T_{select})$ voltage $V_1(t_{n-1} + T_{select})$ equals

$$V_1(t_{n-1} + T_{select}) = \frac{I_{wmax}}{gm}(1 - e^{-\frac{T_{select}}{\tau}}) \tag{E.11}$$

Time constant τ equals $\tau = \frac{C+C_w}{gm}$. At the end of phase ϕ_1, switch S_1 is opened and the charge on C_w is preserved. At the beginning of phase ϕ_2, the charge on C_w redistributes on C_w and C_2. Current memory number 2 is initialized at a current $-I_{wmax}$. The voltage at time $t_n + T_{select}$ becomes

$$V_1(t_n + T_{select}) = \frac{I_{wmax}}{gm}(-1 + e^{-\frac{T_{select}}{\tau}} + \frac{C_w}{C_w + C}(e^{-\frac{T_{select}}{\tau}} - e^{-\frac{2T_{select}}{\tau}})) \tag{E.12}$$

When $T_{select} \gg \tau$, the part with the double exponent ($e^{-\frac{2T_{select}}{\tau}}$) may be neglected. The relative settling error (ϵ_s) then becomes

$$\epsilon_s = (1 + \frac{C_w}{C_w + C})e^{-\frac{T_{select}}{\tau}} \tag{E.13}$$

E.3 Impedance ratio errors

In this section, errors by the impedance ratio between the memory cell's output impedance and neurons input impedance are considered.

E.3.1 Impedance ratio errors from synapse outputs to neuron inputs

In larger neural networks, one neuron is fed by a large number of synapses. The synapses are connected to, respectively disconnected from the neuron as a function of the input pattern. The input pattern controls the current steering switched, that are closed in phase ϕ_m^n. The network is time variant. The non-zero output conductance of the synapses cause transfer errors of the effective input current $I_{eff}^n(t)$. The situation is illustrated in figure E.7.

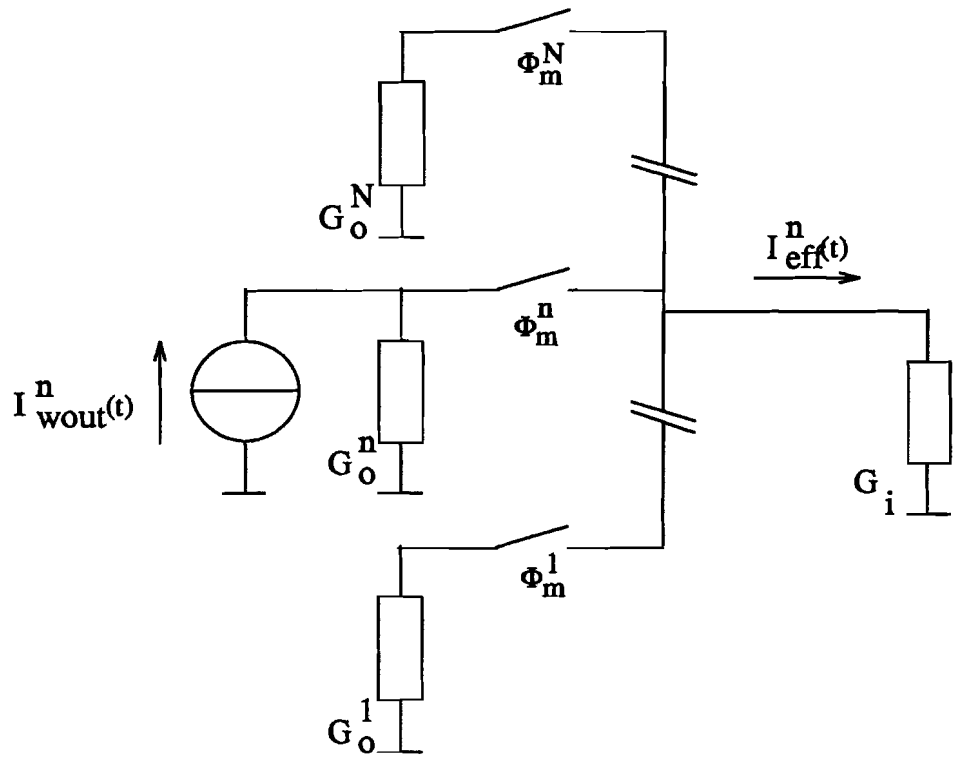

Figure E.7: Impedance ratio errors.

The output impedances of N synapses are considered equal (G_o). The effective input current is given by formula E.14.

$$I^n_{eff}(t) = I^n_{wout}(t) \frac{1}{1 + \frac{\sum_{n=1}^{N} Q(\phi^n_m) G_o}{G_i}} \tag{E.14}$$

with:

$$S(\phi) = \begin{cases} 1 & \text{if } \phi = \text{H} \\ 0 & \text{if } \phi = \text{L} \end{cases} \tag{E.15}$$

The function $Q(\phi)$ signals the multiplying phase for each input signal. The integrators input conductance is denoted as G_i. Each weight current $I^n_{wout}(t)$ is scaled with a time variant factor to $I^n_{eff}(t)$. The multiplication result is the integrated effective input current during the multiplying phase (ϕ^n_m). The time variant factor is only depending on the multiplying phases ϕ^n_m controled by input signals. The time variant scaling factor is a unique factor for each input pattern. It can therefore be regarded as a scaling factor of the input pattern. This scaling of the input pattern is compensated for by the training algorithm.

E.4 Noise errors

The noise of the MOS-transistor may be divided into '1/f-noise' or flicker noise and thermal noise. The flicker noise is a low frequency noise signal, and is therefore largely compensated by the memory cell: The memory cell samples the noise signal in one phase. In the next phase, the low frequency components of the transistor noise signal approximate the low frequency

components of the sampled noise signal. The flicker noise component is compensated in this way. The expected thermal noise power spectrum of a MOS transistor is given by formula E.16. The noise scheme of figure E.8 is used.

$$\overline{I_{th}^2}(f) = \frac{8}{3} m_{th} k T_j g m \Delta f \tag{E.16}$$

The factor $m_{th} \approx 2.5$ is a thermal noise process parameter. T_j is the absolute junction temperature, k is Boltzmann's constant. The thermal noise power spectrum is flat up to very high frequencies.

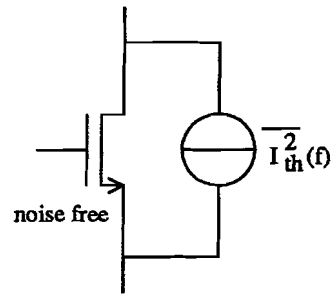

Figure E.8: N-MOS expected noise power scheme.

The bandwidth of the memory cell depends on the state of the switches. In this analysis, the noise power per unit frequency is considered, so the bandwidth is not needed and the noise power is not calculated. The noise output power per unit frequency of the configuration of figure E.1 consist of contributions of all transistors. The contributing noise signals are not correlated, and their noise power contributions can therefore be summed. The noise spectrum of transistor $M1$, $M2$ and current source J are directly summed to the output. When a zero weight situation ($I_{win} = 0A$) is assumed, the noise spectrum of the current source $2J$ is divided equally into the two branches of the circuit:

$$\overline{I_{th}^2 tot}(f) = \frac{8}{3} m_{th} k T_j \Delta f (g m_1 + g m_2 + g m_J + \frac{1}{2} g m_{2J}) \tag{E.17}$$

This situation is illustrated in figure E.9.

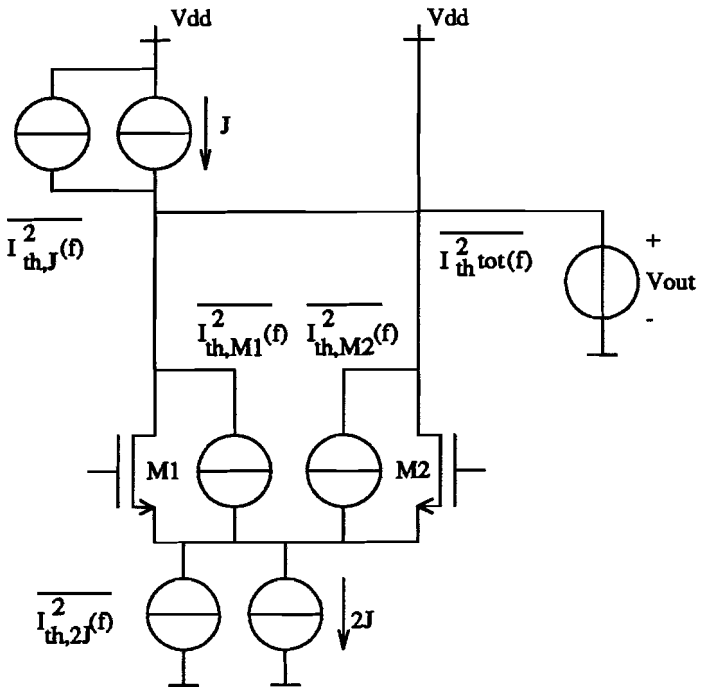

Figure E.9: Current memory expected noise power scheme.

From formula E.17 it clear that the output noise power can be minimized by minimizing the transconductances of the circuits transistors.

Appendix F

High level system requirements

F.1 Integrator requirements

In case of the integrator, the following high level system requirements are taken into account: the clamping voltage, Slew rate of the operational amplifier, its gain-bandwidth product.

F.1.1 Clamping voltage

The integrator output voltage V_{uo+} is clamped at the voltage $+V_{clamp}$ or $-V_{clamp}$. The activation function has reached its extrema at the voltage $+V_{act_{max}}$ and $-V_{act_{max}}$. In order to prevent distortion of the multiplication result, the integrator output voltage has to be prevented from clamping if it is not desired. The worst case trajectory (T_1) is depicted in figure F.1.

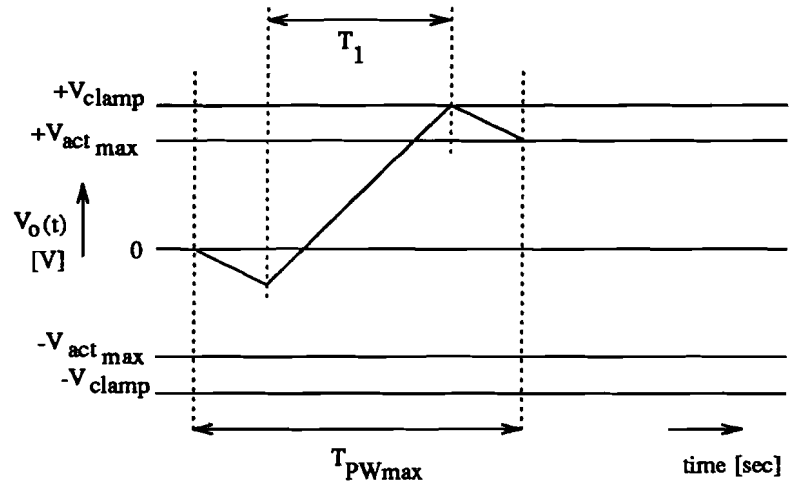

Figure F.1: Worst case output voltage trajectory.

This situation occurs when N_1 of the total of N weight output currents with value I_{wmax} and pulse width T_{PWmax} are applied, and $N - N1$ weight currents with value $-I_{wmax}$ and pulse width T_1. As the trajectory is determined, N_1 can be calculated from the integration capacitance (C_I) and the quantities mentioned above:

$$N_1 = \frac{2C_I(V_{clamp} - V_{act_{max}})}{I_{wmax}(T_{PWmax} - T_1)} \tag{F.1}$$

Although N_1 is a non-integer value in this equation, this value will be used as an approximation. The total current sourced to the integrator during the middle part of the trajectory (I_{totm}) equals

$$I_{totm} = I_{wmax}(2N_1 - N) \tag{F.2}$$

During this middle part, the voltage increases by

$$-(2V_{clamp} - V_{act_{max}}) = \frac{T_1 I_{wmax}}{C_I}(2N_1 - N) \tag{F.3}$$

The quantity N_1 can be substituted from formula F.1. This equation can be solved for the maximum weight current:

$$I_{wmax} = \frac{4C_I(V_{clamp} - V_{act_{max}})}{N(T_{PWmax} - T_1)} + \frac{C_I(2V_{clamp} - V_{act_{max}})}{NT_1} \tag{F.4}$$

This maximum weight current is still a function of T_1, but a minimum value for I_{wmax} can easily be calculated analytically by taking the derivative dI_{wmax}/dT_1, and solving it for T_1 : $dI_{wmax}/dT_1 = 0$:

$$T_1 = \frac{-T_{PWmax}(2V_{clamp} - V_{act_{max}}) + T_{PWmax}\sqrt{8V_{clamp}^2 - 6V_{clamp}V_{act_{max}} + V_{act_{max}}^2}}{2V_{clamp}} \tag{F.5}$$

This value for T_1 can be substituted into formula F.4, in order to obtain the most severe constraint for I_{wmax}. The constraint is checked for the system values of section 5.1.2 and $V_{act_{max}} = 0.5V$ in table F.1. In this case, the following values occur: $T_1 = 160nsec$ and $N_1 = 2.018$

Table F.1: Constraint maximum weight.

Integrator clamping circuitry.				
Formula	*required*	*condition*	*calculated*	*unit*
(F.4)	5	<	6.6	μA

F.1.2 Slew rate

The integrator's opamp must be able to track the voltage across the integration capacitance (C_I). The required slew rate (SR) of the opamp in [V/s] equals

$$SR \geq N\frac{I_{wmax}}{C_I} \tag{F.6}$$

F.1.3 The gain-bandwidth product

When a maximum input current step ($Inet_j(t) = NI_{wmax}\,1(t)$) is applied to the virtual ground node, the voltage of the virtual ground has to remain within the input range of the opamp and the memory cell (CMIR=$1V$). In order to calculate the modulation of the input node, this voltage is calculated back from the output of the integrator (figure 6.6) The ideal integrator output signal is assumed to be

$$V_{out}(t) = \frac{NI_{wmax}}{C_I} r(t) \tag{F.7}$$

with $r(t)$ a ramp function from $t = 0$. In the s-domain, this output signal can be written as

$$V_{out}(s) = \frac{NI_{wmax}}{C_I s^2}. \tag{F.8}$$

The voltage of the virtual ground node can be calculated back:

$$V_{in}(s) = -\frac{NI_{wmax}}{A_0 C_I s^2}(1 - s/p_0) \tag{F.9}$$

The open DC loop gain of the opamp is denoted as A_0, and p_0 is the dominant pole of the opamp. This function can be rewritten in the time-domain by

$$V_{in}(t) = -\frac{NI_{wmax}}{A_0 C_I}(r(t) - 1/p_0 1(t)) \tag{F.10}$$

with $1(t)$ the unity step function. Now, the output signal is assumed to clamp at a voltage of V_{clamp}. The input signal has two extrema:

$$\begin{aligned} V_{in}(0) &= \frac{NI_{wmax}}{A_0 p_0 C_I} \\ V_{in}(t_{max}) &= -\frac{NI_{wmax}}{A_0 C_I}(t_{max} - 1/p_0) \end{aligned} \tag{F.11}$$

The second extreme occurs when the output voltage clamps: $t_{max} = \frac{V_{clamp} C_I}{NI_{wmax}}$. In this case, the opamp is in its unity feed back mode, and $V_{in}(t)$ settles to zero. The extrema of formula F.11 have to input range of the opamp. This sets requirements for the gain-bandwidth product ($\|A_0 p_0\|$) of the opamp.

F.1.4 Settling time of integrator

The settling behavior of the integrator is calculated from figure F.2. The conductance and the current of the input source ($g_{net_j}(t)$ and $I_{net_j}(t)$) are time dependent because of the varying number of memory cell's connected to the integrator.

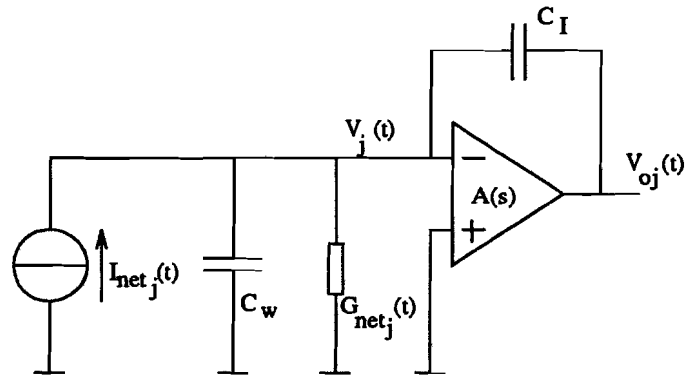

Figure F.2: Scheme for settling behavior integrator.

In the following analysis, $G_{net_j}(t)$ and $I_{net_j}(t)$ are assumed to be constant: $G_{net_j}(t) = G_{net_j}$ and $I_{net_j}(t) = I_{net_j}$. Further more, the opamp is assumed to be a first order system: $A(s) = \frac{A_0}{(1 - s/p_0)}$,

with A_0: the opamp's DC-gain, and p_0: the opamp's dominant pole. The output voltage to input current transconductance becomes

$$\frac{V_{o_j}(s)}{I_{net_j}(s)} = -\frac{1}{\frac{G_{net_j}}{A_0}(1 - s/p_0)(1 - s/p_1) + sC_I} \tag{F.12}$$

with $p_1 = -\frac{G_{net_j}}{C_w + C_I}$, the pole of the feedback loop. The system is of second order. When the poles of the system are assumed to be widely spaced, the poles are:

$$\begin{aligned} p_A &= -\frac{1}{-\frac{1}{p_0} - \frac{1}{p_1} + \frac{A_0 C_I}{G_{net_j}}} \\ p_B &= p_0 + p_1 - \frac{p_0 p_1 A_0 C_I}{G_{net_j}} \end{aligned} \tag{F.13}$$

In normal operation is pole $\|p_1\| \gg \|p_0\|$. This implies that $\|p_B\| \gg \|p_A\|$. This means that pole p_A is the wanted integrator pole, and p_B the pole that determines the integrator settling time $T_{s_{int}}$:

$$\epsilon_{s_{int}} = e^{(p_0 + p_1 - \frac{p_0 p_1 A_0 C_I}{G_{net_j}})T_{s_{int}}} \tag{F.14}$$

with $\epsilon_{s_{int}}$ the integrator settling error.

F.2 Comparator requirements

In order to obtain a pulse width modulated signal of sufficient quality, the comparator's slew rate must be high enough. The smallest pulse width is $T_{PW_{min}}$. In order to produce a pulse with this width, the comparator's output signal must have switched within $T_{PW_{min}}$ seconds as illustrated in figure F.3.

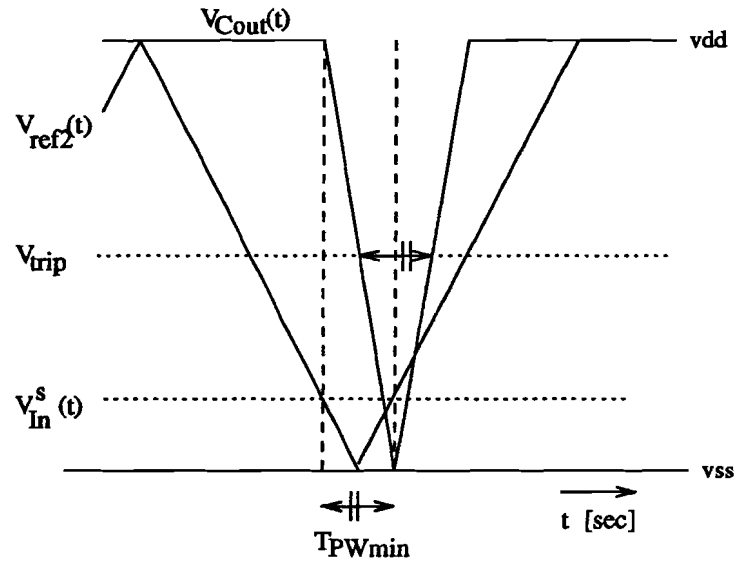

Figure F.3: Minimum comparator slew rate calculation.

In this figure, the slew rate of the inverters is negelected. The voltage of the output node of the comparator ($V_{Cout}(t)$) must have reached ground at the moment that $V_{ref2}(t) = V_{In}(t)$. The minimum pulse width (T_{PWmin}) will not be achieved otherwize. When the slew rate of increasing voltages is different from the slew rate of decreasing voltages, the smallest comparator slew rate (SR_{comp}) must be:

$$SR_{comp} \geq \frac{vdd}{T_{PWmin}} \tag{F.15}$$

Appendix G

Dimensioning the neural hardware

G.1 Dimensioning the differential memory cell

The dimensioning of the memory cell (figure G.1) involves a number of important requirements as shown in chapter 5. The calculations that are made on this topology are summarized below.

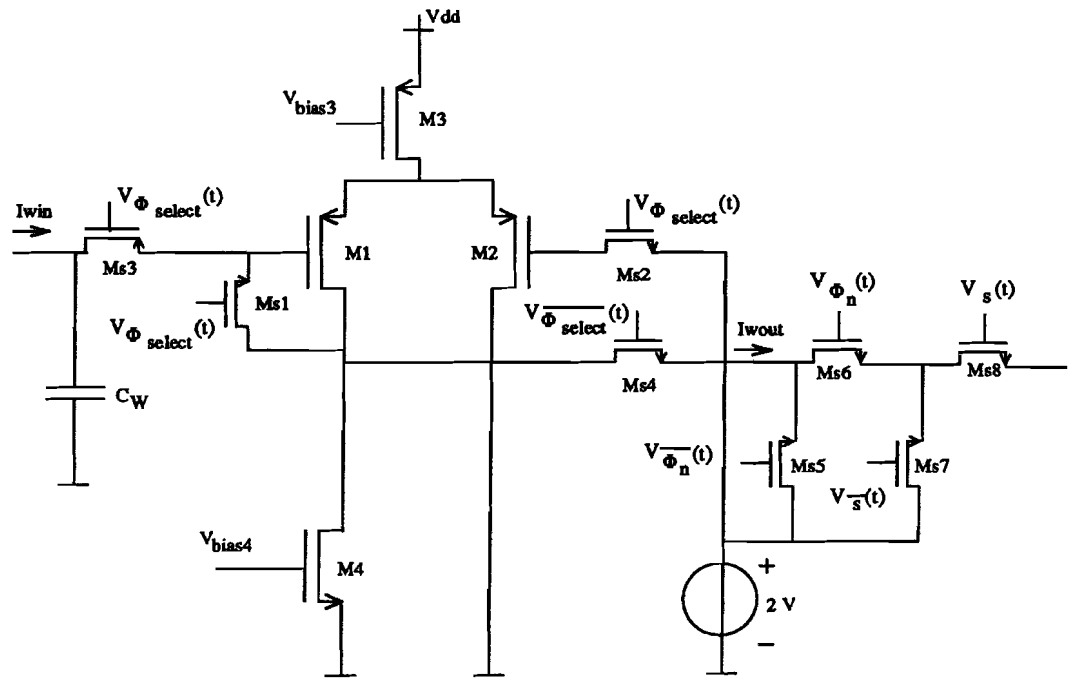

Figure G.1: P-type memory cell.

G.1.1 DC-requirements

The output common mode range (CMOR) is set to $1V$. The body effect is not taken into account fro simplicity. At the lower side, transistor $M4$ has to remain in saturation:

$$\sqrt{\frac{2I_{wmax}}{K_4}} < Vref - \frac{CMOR}{2} \tag{G.1}$$

At the higher side, transistor $M1$ and $M3$ must remain in saturation. At the worst case situation $I_{dM1} = 2I_{wmax}$:

$$Vdd - \sqrt{\frac{4I_{wmax}}{K_3}} - \sqrt{\frac{4I_{wmax}}{K_1}} > Vref + \frac{CMOR}{2} \tag{G.2}$$

G.1.2 Dynamical-requirements

The dominant time constant of the memory cell occurs in the sampling phase. Figure G.2 shows the diode connected memory cell. The switches on resistance is assumed to be zero. The transistors output conductance g_{ds} and source to bulk transconductance gm_{bs} are not taken into account.

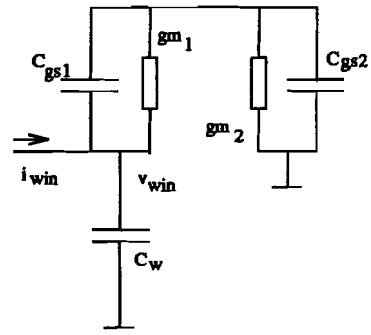

Figure G.2: Memory cell first order small signal scheme.

In order to calculate the dominant time constant, the zero weight situation is considered: $C_{gs1} = C_{gs2} = C_{gs}; gm_1 = gm_2 = gm$. The impedance seen from the input node, then becomes:

$$\frac{v_{win}(s)}{i_{win}(s)} = \frac{\frac{2}{gm}}{1 + s\frac{2C_w + C_{gs}}{gm}} \tag{G.3}$$

The requirement for the time constant (τ) of this impedance were derived in chapter 5:

$$\tau = \frac{2C_w + C_{gs}}{gm} < \frac{-T_r}{ln(2^{-8}/2)} \tag{G.4}$$

G.2 Dimensions and constraints of differential memory

The dimensions of the transistors are given in Table G.1.

Table G.1: Dimensions differential memory cell

Differential memory cell			
Transistor	*W*	*L*	*unit*
M_1	42	7.2	μm
M_2	42	7.2	μm
M_3	24	4.8	μm
M_4	4.8	24	μm

The constraints of the memory cell are summarized in Table G.2.

Table G.2: Constraints differential memory cell

Differential memory cell				
Formula		*condition*		*unit*
(G.1)	1.45	<	1.50	*V*
(G.2)	4.06	>	2.50	*V*
(G.4)	37	<	80	*nsec*

G.3 Dimensioning the integrator

The complete integrator scheme is used, as shown in figureG.3. At first the DC-requirements of the opamp will be addressed.

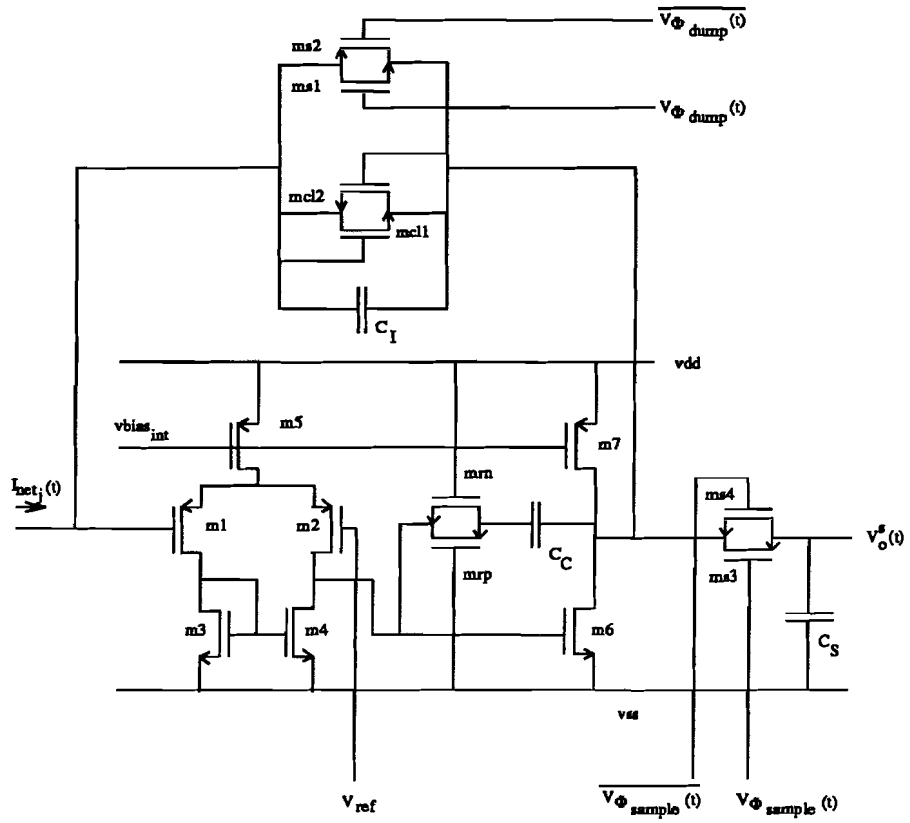

Figure G.3: Complete integrator scheme.

G.3.1 DC-requirements

The common mode input range (CMIR) is set to $1V$. This equals the output common mode range of the memory cell. For simplicity, the body effect is not taken into account. At the higher side, transistor $M5$ has to remain in saturation:

$$Vref + \frac{CMIR}{2} < Vdd + Vtp - \sqrt{\frac{2I_{M5}}{K_1}} - \sqrt{\frac{2I_{M5}}{K_5}} \tag{G.5}$$

At the lower side, transistor $M1$ has to remain in saturation:

$$Vref - \frac{CMIR}{2} > Vtp + Vtn + \sqrt{\frac{2I_{M5}}{K_3}} \tag{G.6}$$

At the $M4$ side of the differential pair, the gate-source voltage of $M6$ has to be pulled sufficiently high to let $M6$ source $2I_{M7}$:

$$Vref - \frac{CMIR}{2} > Vtp + Vtn + \sqrt{\frac{4I_{M7}}{K_6}} \tag{G.7}$$

The opamp's output range (OR) has to be large enough to provide the required voltage sweep of $2V_{clamp}$. At the lower side, $M6$ has to remain saturated:

$$Vref - V_{clamp} > \sqrt{\frac{4I_{M7}}{K_6}} \tag{G.8}$$

At the higher side, $M7$ has to remain saturated:

$$Vref + V_{clamp} < Vdd - \sqrt{\frac{2I_{M7}}{K_7}} \tag{G.9}$$

G.3.2 Dynamical-requirements

The dynamical requirements determine the bias currents (I_{M7}; I_{M5}) of the opamp. The required slew rate (SR) of the opamp depends on the integration capacitance and the maximum integrator input current:

$$SR_{req} = \frac{NI_{wmax}}{C_I} \tag{G.10}$$

The integrator bias currents are determined by these output currents. The output stage bias current has to be large enough to source the loading capacitances ($C_I + C_C + C_s$):

$$\frac{I_{M7}}{C_I + C_C + C_s} > SR_{req} \tag{G.11}$$

The input stage bias current (I_{M5}) has to be able to source the current through the compensation capacitance (C_C). The required slew rate on the gate of transistor $M6$ equals SR_{req}/A_2, with A_2 the voltage gain of the opamp output stage. The miller effect however, enlarges the compensation capacitance by a factor of $(A_2 + 1)$, so the input stage bias current has to be able to source

$$\frac{I_{M5}}{(A_2 + 1)C_C} > \frac{SR_{req}}{A_2} \tag{G.12}$$

For large values of A_2, the relation approximates to

$$\frac{I_{M5}}{C_C} > SR_{req} \tag{G.13}$$

In order to achieve monotonic settling, the dynamical properties of the opamp have to be calculated. The second order simplified small signal equivalent circuit [26] of figure G.4 can be used here.

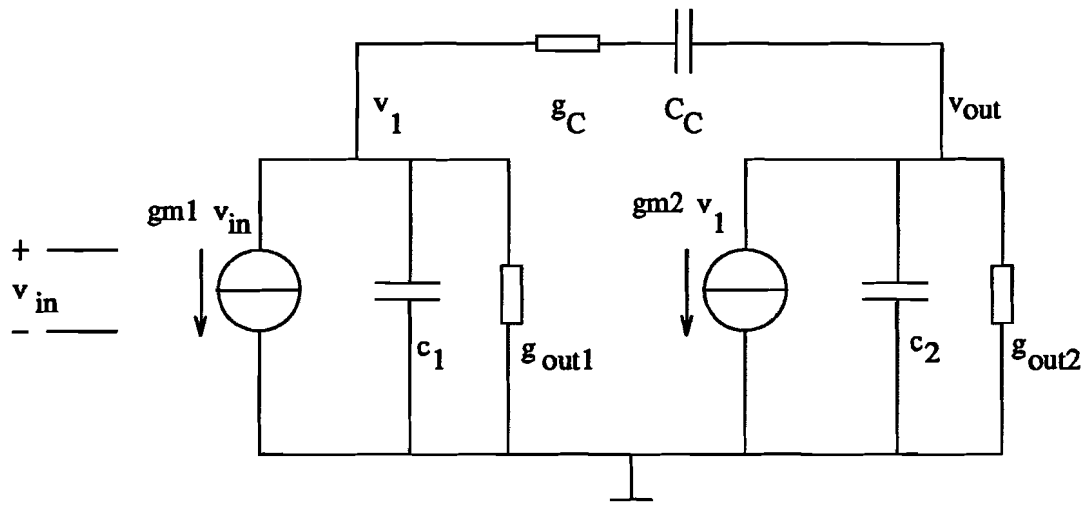

Figure G.4: Integrator opamp small signal equivalent circuit.

The circuit has three poles and one zero. The poles and zeros can be approximated by

$$\begin{aligned} p_1 &\approx -\frac{g_{out1}g_{out2}}{gm2C_C} \\ p_2 &\approx -\frac{gm2}{C_2} \\ p_3 &\approx -\frac{1}{R_C C_1} \\ z_1 &\approx \frac{1}{C_C(1/gm2 - R_C)} \end{aligned} \tag{G.14}$$

Pole p_1 is dominant and zero z_1 is located in the right-half-plane. This zero reduces the phase margin. It can be shifted to the left half plane by taking

$$R_C > 1/gm2 \tag{G.15}$$

In order to achieve monotonic settling, the pole p_2 has to be $4(A_0 + 1)$ times larger than the dominant pole. A_0 is the opamp's DC-gain:

$$A_0 = \frac{gm1gm2}{g_{out1}g_{out2}} \tag{G.16}$$

When the small signal parameters of the transistors of figure G.3 are used, the poles and zeros become

$$\begin{aligned} p_1 &\approx -\frac{(gds_1+gds_3)(gds_6+gds_7)}{gm_6 C_C} \\ p_2 &\approx -\frac{gm_6}{(C_I+C_S)} \\ p_3 &\approx -\frac{(gds_{rn}+gds_{rp})}{Cgs_6} \\ z_1 &\approx \frac{1}{C_C(1/gm_6 - 1/(gds_{rn}+gds_{rp}))} \end{aligned} \tag{G.17}$$

The opamp's DC-gain equals

$$A_0 = \frac{gm_1 gm_6}{(gds_1 + gds_3)(gds_6 + gds_7)} \tag{G.18}$$

In order to achieve monotonic settling, the following constraint has to be fulfilled:

$$|p_2| > 4(A_0 + 1)|p_1| \tag{G.19}$$

(G.20)

The modulation of the voltage of the virtual ground node must remain within the common mode input range (CMIR) of the opamp and the memory cell. The extrema of the input nodes's voltage are derived in formula F.11. The constraints involved are:

$$|\frac{NI_{wmax}}{A_0 p_1 C_I}| < \frac{CMIR}{2} \tag{G.21}$$

$$|\frac{NI_{wmax}}{A_0 C_I}(\frac{V_{clamp}C_I}{NI_{wmax}} - 1/p_1)| < \frac{CMIR}{2} \tag{G.22}$$

G.4 Dimensions and constraints of the integrator

The dimensions of the transistors can be calculated when the integration capacitance ($C_I = 2pF$) is chosen. The required slew rate (SR_{req}) then will be $SR_{req} = 20V/\mu s$. The dimensions summarized below (Table G.3) are chosen by calculations through hand and by simulation.

Table G.3: Dimensions integrator

Integrator			
Element	*W or other*	*L*	*unit*
M_1	9.6	4.8	μm
M_2	9.6	4.8	μm
M_3	4.8	4.8	μm
M_4	4.8	4.8	μm
M_5	16	2.4	μm
M_6	96	2.4	μm
M_7	96	2.4	μm
M_{rn}	4.8	4.8	μm
M_{rp}	12	4.8	μm
M_{cl1}	64	2.4	μm
M_{cl2}	64	2.4	μm
M_{s1}	32	2.4	μm
M_{s2}	32	2.4	μm
M_{s3}	2.4	2.4	μm
M_{s4}	2.4	2.4	μm
C_I	2.0	-	pF
C_C	0.4	-	pF
C_S	0.2	-	pF
I_{M5}	10	-	μA
I_{M7}	60	-	μA

The constraints that are explained above are summarized in Table G.4.

Table G.4: Constraints integrator

Integrator				
Formula		*condition*		*unit*
(G.5)	2.5	<	2.95	V
(G.6)	1.5	>	0.65	V
(G.7)	1.5	>	0.35	V
(G.8)	0.70	>	0.37	V
(G.9)	3.30	<	4.58	V
(G.11)	23	>	20	$V/\mu sec$
(G.13)	25	>	20	$V/\mu sec$
(G.15)	3.26	>	2.04	$k\Omega$
(G.19)	393	>	183	$Mrad/sec$
(G.21)	0.438	<	0.5	V
(G.22)	0.439	<	0.5	V

G.5 Dimensioning the comparator

The complete comparator scheme is used, as shown in figure G.5. At first the DC-requirements of the comparator will be addressed.

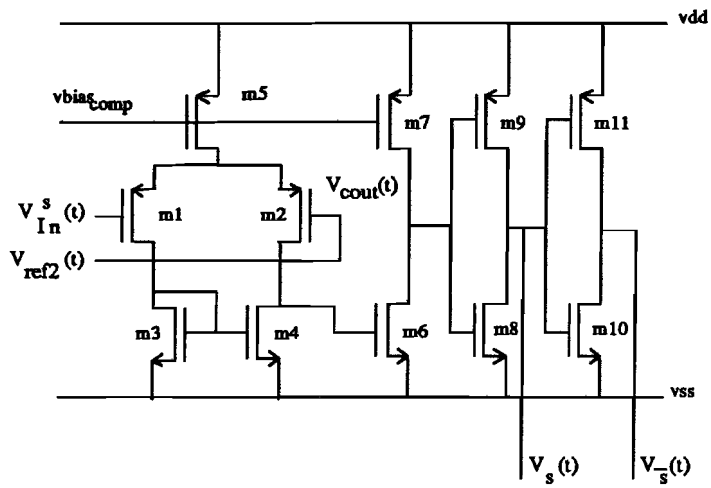

Figure G.5: Comparator scheme.

G.5.1 DC-requirements

The common mode input range (CMIR) is set to $CMIR = 2V_{clamp}$. This equals the output common mode range of the integrator. For simplicity, the body effect is not taken into account. At the higher side, transistor $M5$ has to remain in saturation:

$$Vref + \frac{CMIR}{2} - Vtp + \sqrt{\frac{2I_{M5}}{K_1}} < Vdd - \sqrt{\frac{2I_{M5}}{K_5}} \tag{G.23}$$

At the lower side, transistor $M1$ has to remain in saturation:

$$Vref - \frac{CMIR}{2} > Vtp + Vtn + \sqrt{\frac{2I_{M5}}{K_3}} \tag{G.24}$$

In order to minimize the offset voltage of the comparator caused by channel length modulation, the drain voltages of $M1$ and $M2$ have to be equal at the trip point:

$$Vtn + \sqrt{\frac{I_{M5}}{K_3}} = Vtn + \sqrt{\frac{2I_{M7}}{K_6}} \tag{G.25}$$

G.5.2 Dynamical requirements

Only the large signal dynamical requirements are taken into account. The bias currents I_{M7} and I_{M5} have to be large enough to provide the desired slew rate:

$$\frac{I_{M5}}{C_{gs6}} > \frac{Vref - \frac{CMIR}{2} - Vtp}{T_{PWmin}} \tag{G.26}$$

$$\frac{I_{M7}}{C_{gs8} + C_{gs9}} > \frac{Vdd}{T_{PWmin}} \tag{G.27}$$

G.6 Dimensions and constraints of the comparator

The dimensions of the transistors can be calculated according to the requirements above. The common mode input range ($CMIR$) is set to $CMIR = 2.6V$. The dimensions of the elements are summarized in Table G.5.

Table G.5: Dimensions comparator.

Comparator			
Element	*W or other*	*L*	*unit*
M_1	72	4.8	μm
M_2	72	4.8	μm
M_3	9.6	4.8	μm
M_4	9.6	4.8	μm
M_5	96	2.4	μm
M_6	19.2	4.8	μm
M_7	96	2.4	μm
M_8	16	2.4	μm
M_9	48	2.4	μm
M_{10}	16	2.4	μm
M_{11}	48	2.4	μm
I_{M5}	15	-	μA
I_{M7}	15	-	μA

The constraints that are used in this above are summarized in Table G.6.

Table G.6: Constraints comparator

Comparator				
Formula		*condition*		*unit*
(G.23)	4.49	<	4.79	V
(G.24)	0.70	>	0.55	V
(G.26)	307	>	39	$\frac{V}{\mu sec}$
(G.27)	181	>	125	$\frac{V}{\mu sec}$