

1990

# On solving algebraic equations using analog circuits for nonlinear digital signal processing

Trang D. Nguyen  
*Lehigh University*

Follow this and additional works at: <https://preserve.lehigh.edu/etd>



Part of the [Electrical and Computer Engineering Commons](#)

---

## Recommended Citation

Nguyen, Trang D., "On solving algebraic equations using analog circuits for nonlinear digital signal processing" (1990). *Theses and Dissertations*. 5352.

<https://preserve.lehigh.edu/etd/5352>

This Thesis is brought to you for free and open access by Lehigh Preserve. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of Lehigh Preserve. For more information, please contact [preserve@lehigh.edu](mailto:preserve@lehigh.edu).

**ON SOLVING ALGEBRAIC EQUATIONS**  
**USING ANALOG CIRCUITS**  
**FOR**  
**NONLINEAR DIGITAL SIGNAL PROCESSING**

by  
TRANG D. NGUYEN

A thesis  
Presented to the Graduate Committee  
of Lehigh University  
in candidate for the Degree of  
Master of Science  
in  
Electrical Engineering

Lehigh University

1990

This thesis is accepted in partial fulfillment of the requirements for the degree of Master of science.

5/9/90

---

(date)

*Douglas Gray*

---

Professor in charge

*Lawrence Vanerini*

---

Chairperson of Department

# Acknowledgements

I wish to deeply thank Professor Douglas Frey, my thesis adviser, for his guidance and invaluable advice during the course of this study.

I am indebted to the support and encouragement from my family.

# Table of Contents

<b>abstract</b>	<b>1</b>
<b>1. INTRODUCTION</b>	<b>2</b>
<b>2. Digital processing from nonlinear analog networks -The whole scheme-</b>	<b>5</b>
2.1 MODELING NONLINEAR COMPONENTS AND STANDARD FORM EQUATION FOR NONLINEAR NETWORKS.	7
2.2 DIGITAL PROCESSING FROM NONLINEAR NETWORKS.	12
2.3 CIRCUIT DEMONSTRATION :	18
<b>3. Circuit topologies to realize equation <math>x = w - BF(x)</math> incorporated with DSP network</b>	<b>25</b>
3.1 A HARDWARE APPROACH	27
3.2 FIRST ORDER EQUATION	29
3.2.1 Circuit topology	29
3.2.2 Programming gain B	32
3.3 SECOND AND HIGHER ORDER EQUATIONS	36
3.3.1 Second order equation	36
3.3.2 Higher order equations	42
<b>4. Programmable Polynomial Module incorporated with nonlinear DSP network</b>	<b>46</b>
4.1 THE PRINCIPLE	47
4.2 SQUARE LAW CIRCUIT CONFIGURATION	51
4.3 POLYNOMIAL MODULE	56
4.4 DESIGN CONSIDERATION	59
4.4.1 Voltage-current converter stage	59
4.4.2 Transistor mismatch	62
4.4.3 Maximum input voltage swing	69
4.4.4 Maximum output current	69
4.5 PROGRAMMING COEFFICIENTS $a_i$ 's	70
<b>5. Existence and uniqueness of solution for the equation <math>x=w-BF(x)</math> of nonlinear networks</b>	<b>78</b>
5.1 BACKGROUND	78
5.2 EXISTENCE AND UNIQUENESS OF SOLUTION FOR EQUATION $x=w-BF(x)$	82
5.2.1 Nonlinearities with strictly monotone-increasing characteristics.	82

5.2.1.1 Sufficient condition for the uniqueness of solution for $x=w-BF(x)$	82
5.2.1.2 Necessary and sufficient condition for the uniqueness of solution for equation $x=w-BF(x)$	84
5.2.2 Nonlinearities with non-monotone characteristics	86
<b>6. Experiment and Circuit simulation results</b>	<b>92</b>
6.1 EXPERIMENTAL RESULTS WITH EXPONENTIAL NONLINEARITIES	93
6.2 CIRCUIT SIMULATION RESULTS WITH SQUARE LAW NONLINEAR CHARACTERISTICS	109
6.2.1 First order circuit	109
6.2.2 Second order circuit	112
6.3 POLYNOMIAL MODULE AND SIMULATION RESULTS	116
6.4 POLYNOMIAL WITH PROGRAMMABLE COEFFICIENTS	122
<b>7. Summary and conclusion</b>	<b>128</b>
<b>reference</b>	<b>132</b>
<b>vita</b>	<b>134</b>

# abstract

Recently it has been shown by Professor Frey that by applying a new generalized modeling technique any nonlinear network composed of two-terminal devices can be replaced by an associated linear network L. The structure of the network L together with the models allow the determination of network equations in the standard form  $X = Au + BF(X)$ . Based on the associated network L and its standard form equation, a method to implement a digital equivalent from nonlinear analog networks was developed. The digital equivalent network contains a solve-for-F(X) subnetwork N which solves for F(X) according to equation  $X(n) = w(n) - BF[X(n)]$ . In this work a standard subnetwork N has been designed to operate in the equivalent DSP network. The designed subnetwork N provides means to adjust  $w(n)$  and B depending on the particular application and nonlinear devices. Networks with exponential and polynomial nonlinearities are studied in detail. The concept of a Programmable Polynomial Module is introduced and employed in the standard subnetwork N to implement an arbitrary nonlinear characteristic. Experimental results are shown in comparison to simulation.

# Chapter 1

## INTRODUCTION

Sophisticated signal processing algorithms and hardware are prevalent in a wide range of systems, from highly specialized military systems through industrial applications to low cost, high-volume consumer electronics.

The field of signal processing has always benefited from a close coupling between the theory, applications, and technologies for implementing signal processing systems. Prior to the 1960's, the technology for signal processing was almost exclusively continuous time analog technology. By this time, computers were few, very expensive, not very powerful and reliable and hard to program and use. The rapid evolution of digital computers and microprocessors together with some important theoretical developments caused a major shift to digital technologies, giving rise to the field of digital signal processing.

Digital networks have several advantages over their analog counterparts :

- Higher stability. They are considerably less subject to noise than analog networks.
- More reliable. They are far less sensitive to variation of component parameter values. Simple error detection and correction mechanisms can be used in digital networks.
- Higher maintainability. They need no adjustments and no requirements for trimming and calibration process.
- Lower cost. The advances of microelectronics technology in recent years have made complex systems at very low



cost but its impact has been larger in digital systems. Moreover, a general purpose digital system can be programmed for a particular processing task, eliminating the need to have different systems for each task.

- and In most cases, more accurate than analog systems due to the fact that the infinite small resolution that is theoretically possible in analog area can never be realized because of noise.

Because of above reasons, analog electronics are normally only tolerated in the absence of digital integrated circuits either fast enough to do a task or priced low enough. This critical problems in terms of speed and low operating frequency, however, have been the subject of several current promising researches and the convergence to a better solution is becoming more and more realistic.

Nevertheless, there are circumstances beyond the scope of digital signal processing. How to convert a nonlinear analog network into a digital equivalent is still an unsolved problem.

To digitalize an analog system, depending on particular application one may use one of several available mapping techniques ; e.g Euler, impulse invariant, bilinear etc.; to map a transfer function from S- domain into Z- domain. Inherently, these mapping techniques require a Laplace transfer function of the network must be existed, which is just a matter of algebra for linear analog networks. For nonlinear networks, however, it's imposible to obtain a Laplace transfer function. Thus, so far there is no available method to implement a digital equivalent from nonlinear analog networks.

Extensive works have been devoted to the area of nonlinear

network by Dr. Douglas Frey, a professor of Computer Science and Electrical Engineering Department at Lehigh University. Currently, he is searching a general method to implement digital equivalents from nonlinear analog networks, The basic principles of his method, reported in [1], may open a new direction to complete the picture of digital signal processing area.

It's the interest in his idea has lead me to this study to explore a small part of his idea and hopefully have certain contribution to later researches involve in nonlinear digital signal processing.

In this study the whole scheme is presented in chapter 2. Chapter 3 discusses several circuit topologies to realize the standard equation incorporated with DSP network from nonlinear analog structures. The concept of a Programmable Polynomial Module is introduced and designed in chapter 4 to implement an arbitrary nonlinear characteristic . Chapter 5 derives necessary and sufficient conditions imposed on network topology and properties of nonlinear devices to guarantee the existence of a digital equivalent network. Consecutive chapters report experiment and circuit simulation results to verify theoretical study.

>

## **Chapter 2**

# **Digital processing from nonlinear analog networks**

### **-The whole scheme-**

Although the understanding of nonlinear networks has become more advanced in the past few decades , their applications are only beginning to be explored . An increasing interest in nonlinear networks can be perceived in the signal community by this time. There are considerable works have dedicated to nonlinear networks both in theory and practical applications [2],[3].

Having been classically regarded in the calculation of DC operating points for transistor circuits, or in control and system identification topics, nonlinear structures are now being collected and adapted for signal processing purposes. A good deal of applications on nonlinear structures is being directed to develop synthetic generation mechanisms to evaluate or design processing systems. That is the case of image processing and pattern recognition area where linear filtering methods have proved to be of limited help and nonlinear methods seem to be more adequate and more powerful to solve the problems [4]. Several nonlinear prediction shemes were proposed in [5] and [6].

As a result of increasing applications, nonlinear network has found itself particularly interested in the field of digital signal processing. It's well known that the ultimate performance in speech

processing and digital communication can't be realized with linear time-invariant filters and equalizers. This is just a typical circumstance where nonlinear schemes are so desired. Unfortunately, there is no available general method to implement a digital equivalent from a nonlinear analog network as in the linear case.

The traditional techniques in digital signal processing based strongly on a well-established linear analog network theory. In most of situations, a linear analog network with well-known advantages is replaced by an digital equivalent one without significant impairment. The task of replacement involves mapping the network transfer function from S-domain into Z-domain by one of several available techniques. These traditional techniques can't be applied to nonlinear analog networks where due to their nonlinear nature the Laplace transformation gives no help. Theoretically, one may obtain a transfer function for a given nonlinear network as it was derived in [7] via Fourier-Borel transforms, but it's impractical.

Recently, it was shown by Frey in [1] that by a new approach the problem can be solved and a general method to digitalize nonlinear analog structures is possible. The solution in [1] has a particular importance in that it might be the key idea to the future nonlinear digital signal processing.

## 2.1 MODELING NONLINEAR COMPONENTS AND STANDARD FORM EQUATION FOR NONLINEAR NETWORKS.

Modeling is of fundamental importance in the study of nonlinear network. By a simple generalized modeling technique, a standard form equation was derived in [8] by Frey.

By Frey, any two terminal nonlinear element, either resistive or reactive, can be modeled by using linear components in combination with dependent sources. Two terminal elements are devices which may be characterized simply by a relationship between its two fundamental electrical variables: the voltage  $v$  across its terminals and the current  $i$  that flows through the element from one terminal to the other. For a resistor, the relationship is algebraic, while derivatives are required for the characterization of an inductor or a capacitor. The simplest two-terminal resistors are those in which the relationship between their voltage and current is expressed by specifying the value of one of these variables as a single-valued function of the other variable. This is, by far, the most common situation. A tunnel diode, for example, is a nonlinear resistor that is usually characterized in this manner.

In [8], there are two general models were suggested. In figure 2.1a a current-controlled voltage source and a voltage-controlled current source are used with the linear component  $Z$  and in the figure 2.1b the model has a voltage controlled voltage source and a current-controlled current source.

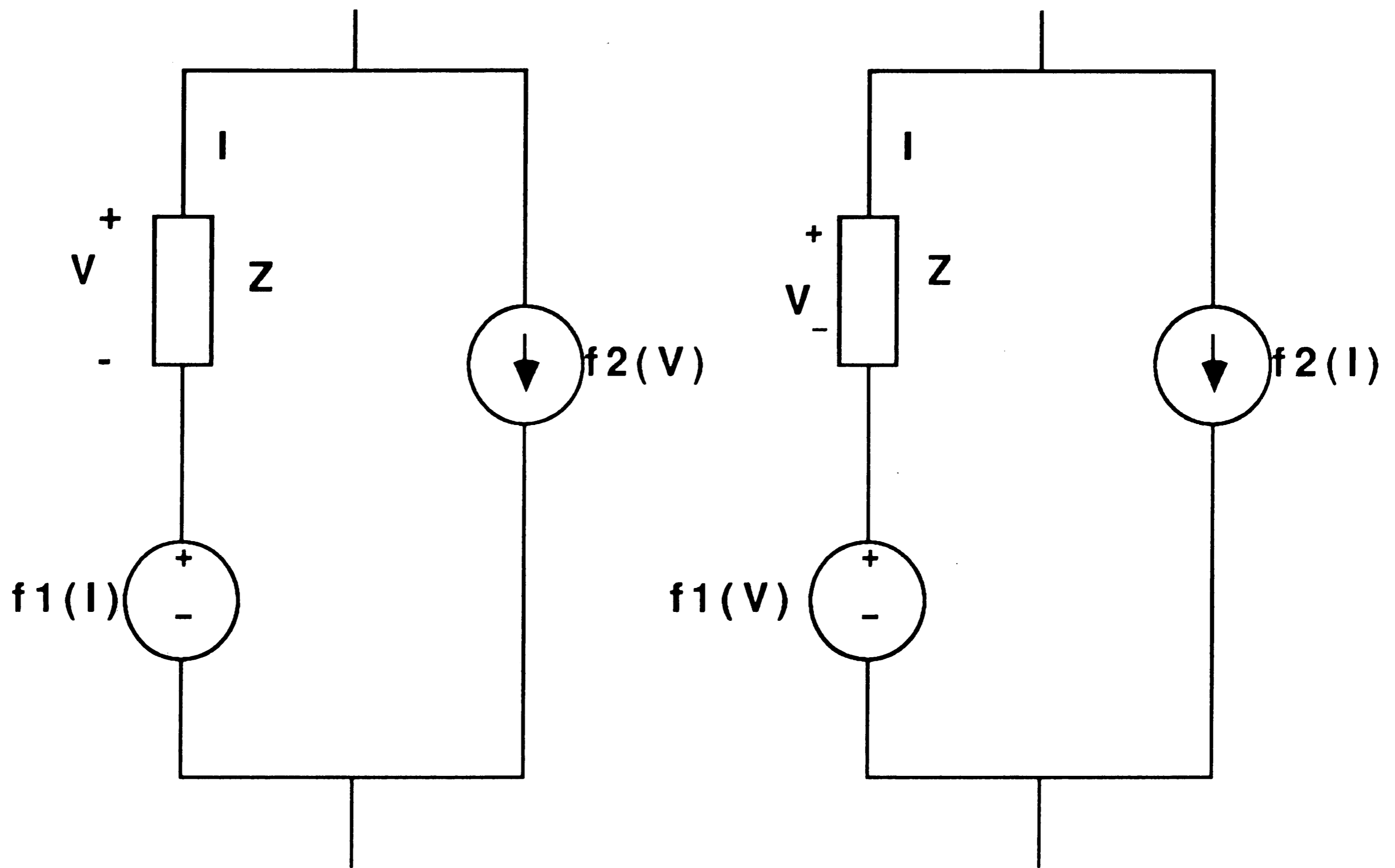


fig. 2.1a

fig. 2.1b

figure 2.1

Generalized model for two-terminal nonlinearity

The voltage and current sources are designated through the functions  $f_1$  and  $f_2$  as shown, where for a given nonlinear component, one of these functions will be constant. In certain cases  $f_1$  and  $f_2$  may be differentiated with respect to time. Due to the simple relationship between voltage and current in a linear resistor both models in figure 2.1 are completely equivalent for the case of nonlinear resistance. Voltage-controlled capacitors and current-controlled inductors can be modeled by using the network of figure 2.1a where  $f_1$  and  $f_2$  are differentiated with respect to time, the model in figure 2.1b is suitable for charge- controlled capacitors and flux-controlled inductors.

Let a nonlinear network  $N$  be given. Assume that  $N$  contains only resistive nonlinearities. We establish the associated linear network  $L$  by replacing all nonlinear components with applicable models. The associated network then is composed of linear components, independent sources and dependent sources.

By treating all nonlinear dependent sources as though they were independent sources we may use the results of linear network analysis to obtain equation:

$$Y.V_N = a_1.u + a_2.F(v) \quad (2.1)$$

Where:

$Y$ : Nodal admittance  $n \times n$  matrix

$V_N$ :  $n$ -vector node voltage.  $n$  is the number of nodes in network excluded datum ( ground node ). For simplicity of notation, all



vectors in this section are column vector .

$a_1$ : Real  $n \times m$  matrix, where  $m$  the is number of independent sources . It has only 1's, -1's, and 0's as entries.

$u$ :  $m$ -vector , stands for independent branch voltage sources and link current sources.

$a_2$ : Real  $n \times k$  matrix, where  $k$  is the number of nonlinearities in network.

$F(v)$ : Dependent sources modeling nonlinearities, a mapping of  $R^k$  into  $R^k$ .

$v$ :  $k$ -vector composed of branch voltages and link currents. The branch voltages are those associated with linear components introduced to the network by using nonlinear models of figure 2.1 , and the link currents are those associated with linear components in the links introduced through the models.  $n$ -vector  $V_N$  and  $k$ -vector  $v$  can be written as:

$$v = M_v \cdot V_N \quad (2.2)$$

From (2.1) :

$$V_N = Y^{-1} \cdot a_1 \cdot u + Y^{-1} \cdot a_2 \cdot F(v) \quad (2.3)$$

And from (2.2):

$$v = M_v \cdot Y^{-1} \cdot a_1 \cdot u + M_v \cdot Y^{-1} \cdot a_2 \cdot F(v) \quad (2.4)$$

Denote

$$A = M_v \cdot Y^{-1} \cdot a_1$$

$$B = M_v \cdot Y^{-1} \cdot a_2$$

Equation (2.4) becomes :

$$v = Au + BF(v) \quad (2.5)$$



In equation (2.5), the matrices  $A$  and  $B$  are either constant or linear operator and are determined by the network topology. These matrices are not affected by changes in the values of the independent sources.  $F(v)$  has the simple form given by :

$$F(v) = [f_1(v_1), f_2(v_2), \dots, f_k(v_k)]^T$$

The standard form of equation (2.5) was derived for the case of purely resistive nonlinear network, which is similar to that derived by Sanberg and Wilson in [9]:

$$AF(x) + Bx = c \quad (2.6)$$

also for the purely resistive networks. It turned out that equation (2.5) is also true for general nonlinear networks composed not only resistive nonlinearities but also reactive nonlinearities, as it was derived by Frey in [8]:

$$x = Au + BF(x) \quad (2.7)$$

Therefore we can consider equation (2.5) or (2.6) as a particular case of generalized equation (2.7). Equation (2.7) is of fundamental importance in the study of nonlinear network. In fact, by using (2.7) a new approach to the periodic steady state problem in nonlinear circuits were developed in [10] which can help to solve the problem in nonlinear circuit simulation, namely finding steady state response without having to integrate through the transient regime.

With the aid of immediate variable  $v$ , we can solve for  $f(v)$  from equation (2.5). Once  $F(v)$  is solved we can calculate all node voltages according to equation (2.3). Output vector  $y$  is easily obtained from node voltage vector by:

$$y = M_y \cdot V_N \quad (2.8)$$

## 2.2 DIGITAL PROCESSING FROM NONLINEAR NETWORKS.

As it has been shown above, the associated network of a given nonlinear network is completely described by equations (2.3) through (2.5) and equation (2.8).

The associated network is depicted in figure 2.2.

To implement the associated network in figure 2.2 into digital equivalent we need some special modification for subcircuit N .

Rewrite equation (2.5):

$$v = Au + BF(v)$$

$$\text{Where } A = M_v \cdot Y^{-1} \cdot a_1$$

$$B = M_v \cdot Y^{-1} \cdot a_2$$

In continuous time domain , we have the Volterra equation:

$$v(t) = \int_0^t a(t-\tau)u(\tau)d\tau + \int_0^t b(t-\tau)f[v(\tau)]d\tau$$

Where  $v(t)$ ,  $a(t)$ ,  $u(t)$ ,  $b(t)$ , and  $f[v(t)]$  are inverse Laplace of  $v(s)$ ,  $A(s)$ ,  $u(s)$ ,  $B(s)$  and  $F[v(s)]$  respectively. In discrete time domain, all convolution integrals are replaced by convolution sums:

$$v(n) = \sum_{k=0}^n a(n-k)u(k) + \sum_{k=0}^n b(n-k)f[v(k)]$$

Or

$$v(n) = \sum_{k=0}^n a(n-k)u(k) + \sum_{k=0}^{n-1} b(n-k)f[v(k)] + b(0)f[v(n)] \quad (2.9)$$

In (2.9) the first term  $a(n)*u(n)$  can be calculated easily since it

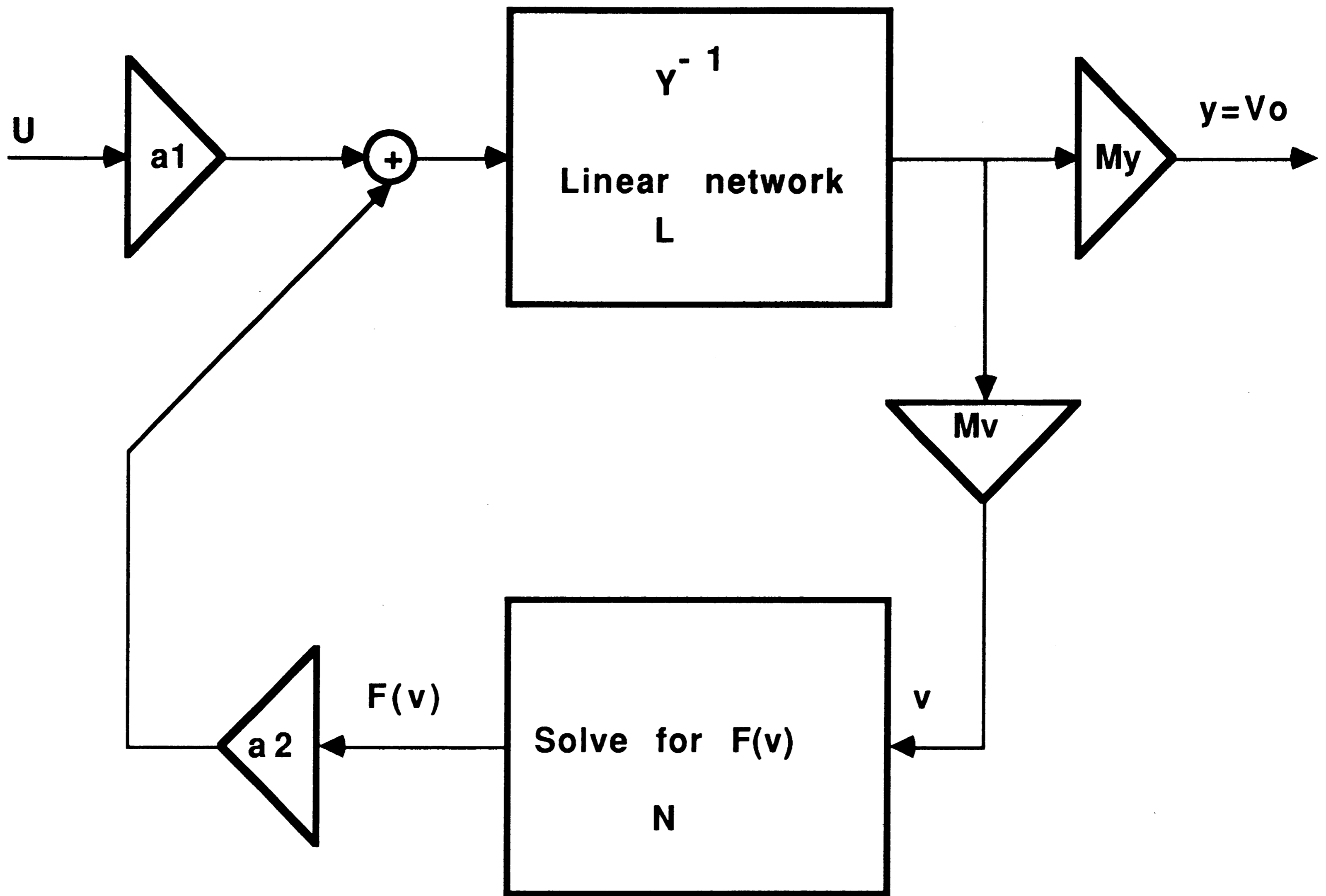


Figure 2.2

The associated network for a given nonlinear network

is the solution of linear problem. The second term represents the past history of the process up to index time (n-1). If at index n, we know all the past values of  $f[v(n)]$  then equation (2.9) enable us to compute the present value of  $f[v(n)]$  at index n .

Denote

$$w(n) = \sum_{k=0}^n a(n-k)u(k) + \sum_{k=0}^{n-1} b(n-k)f[v(k)]$$

$w(n)$  can be written as :

$$w(n) = \sum_{k=0}^n a(n-k)u(k) + \sum_{k=0}^n b(n-k)F^{\wedge}[v(k)] \quad (2.10)$$

Where:

$$F^{\wedge}[v(k)] = \begin{cases} f[v(k)] & \text{if } k < n \\ 0 & \text{if } k = n \end{cases}$$

Equation (2.10) indicates that  $w(n)$  is the result of two discrete-time convolutions. These two convolutions are the outputs of two linear time-invariant systems  $a(n)$  and  $b(n)$  . Where  $a(n)$  and  $b(n)$  are determined entirely by the given network topology.

Then

$$v(n) = w(n) + b(0).F[v(n)] \quad (2.11)$$

To see how equation (2.11) works, let begin at starting point,  $n = 0$ , we have :

$$w(0) = a(0).u(0)$$

and

$$v(0) = w(0) + b(0).F[v(0)]$$

From this equation we can solve for  $F[v(0)]$

At  $n = 1$  :

$$w(1) = a(1).u(0) + a(0).u(1) + b(1).F[v(0)]$$

and :

$$v(1) = w(1) + b(0).F[v(1)]$$

This equation in turn used to solve for  $F[v(1)]$ .

The process repeats in this manner.

In general, the subcircuit N receives internal input  $w(n)$  and generates output  $F[v(n)]$  according to equation (2.11) Internal input  $w(n)$  is determined by linear networks  $a(n)$ ,  $b(n)$  and also by the past values of  $F[v(n)]$  as indicated in equation (2.10)

The question now is how to get a correct value of  $w(n)$  at index  $n$ . This can be solved by note that from equation (2.10):

$$w(n) = a(n) * u(n) + b(n) * F^{\wedge}[v(n)] \quad (2.12)$$

Where  $*$  denotes the convolution sum in discrete time domain.  $F^{\wedge}[v(n)]$  is equal to  $F[v(n)]$  at all index  $k \leq n-1$

Equation (2.12) implies that we can solve the problem by forcing output of subcircuit N to zero at index  $n$  as shown in figure 2.3

To make the network in figure 2.3 work properly, we need to initialize switch S at position 1 at every index  $n$ . The switch S has to switch back to its normal position, position 2, before the next index  $(n+1)$  begins. At index  $(n+1)$  switch S switches to position 1 again and the cycle is repeated. At index  $n$  the past values of  $F[v(n)]$ , i.e  $F^{\wedge}[v(n)]$ , enter linear filter  $b(n)$ , the output of linear filter  $b(n)$  is added to the output of linear filter  $a(n)$  to form internal signal  $w(n)$ . The internal signal  $w(n)$  is then received by sub-network N to

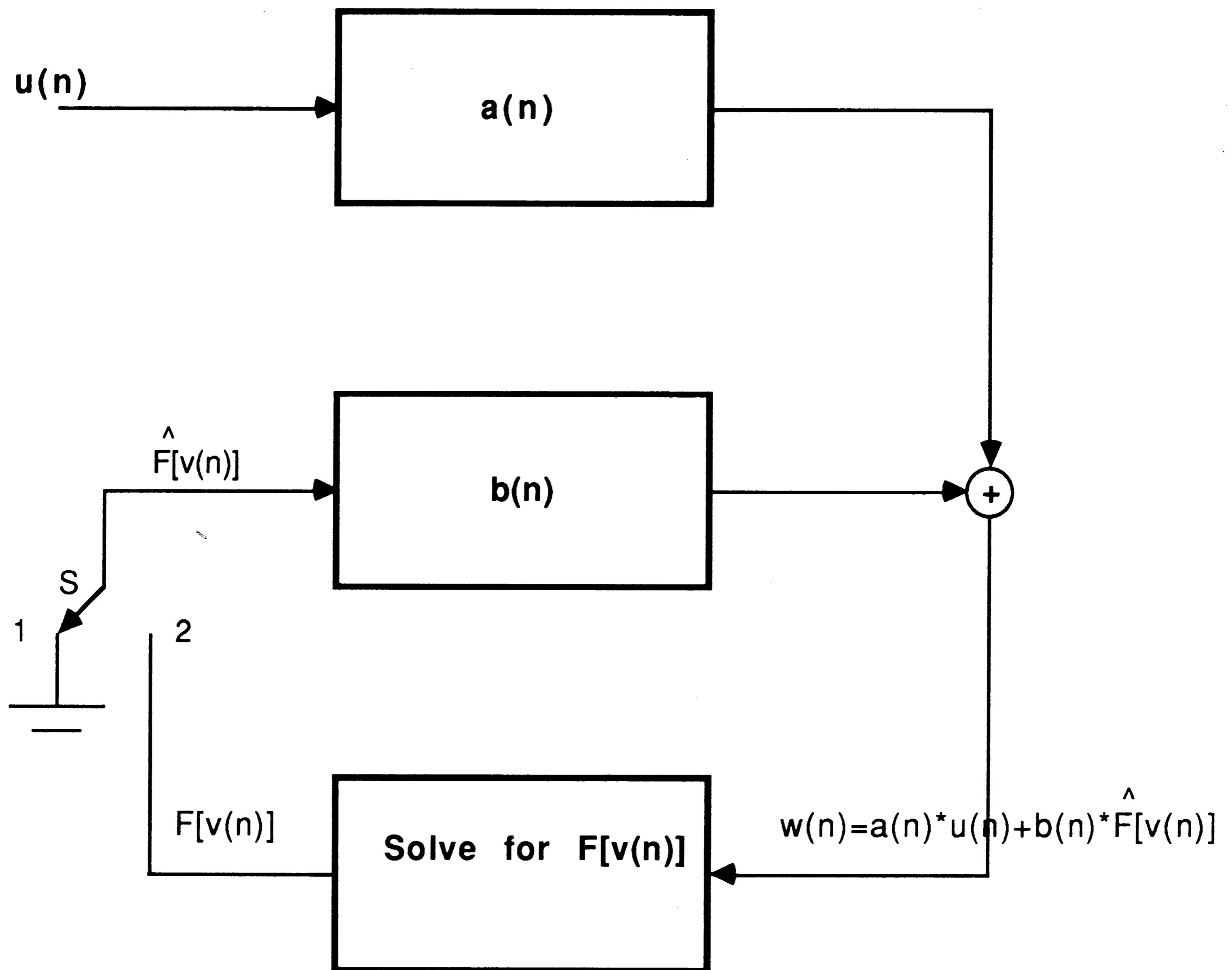


Figure 2.3

**Digitalized Solve-for- $F[v(n)]$  Circuit**

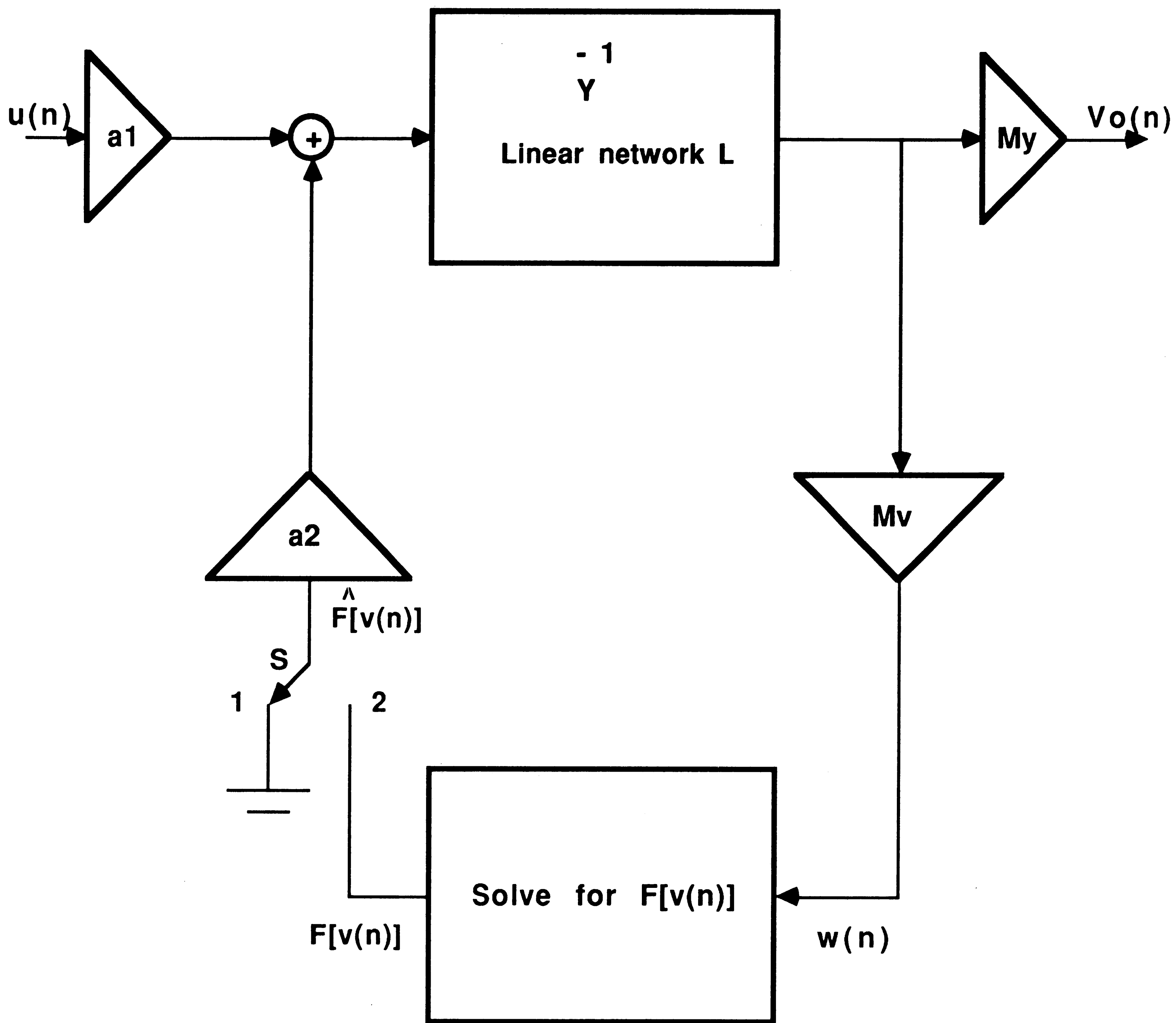


Figure 2.4

Digital equivalent network

generate  $F[v(n)]$ . This new value of  $f[v(n)]$  will update the linear filter  $b(n)$  and an up-to-date signal will appear at output of  $b(n)$ . This updated signal in turn will update consecutive stages of the non-linear DSP network.

In summary, by using a generalized modeling technique, any given nonlinear analog network  $N$  can be replaced by an associated linear network  $L$ . All of the topological properties of  $N$  are preserved in  $L$ . A digital equivalent network for the associated network  $L$  then can be obtained as it was shown in the above procedure. The digital equivalent network for the associated network of figure 2.2 is shown in figure 2.4.

Next section demonstrates how the process works by a simple circuit with one nonlinear component.

### **2.3 CIRCUIT DEMONSTRATION :**

Let consider a simple circuit with a diode used as a nonlinear element in figure 2.5a. By modeling diode  $D$  with a linear resistor  $r$  and a voltage-controlled current source  $f(v)$ , we have the associated network in figure 2.5b.



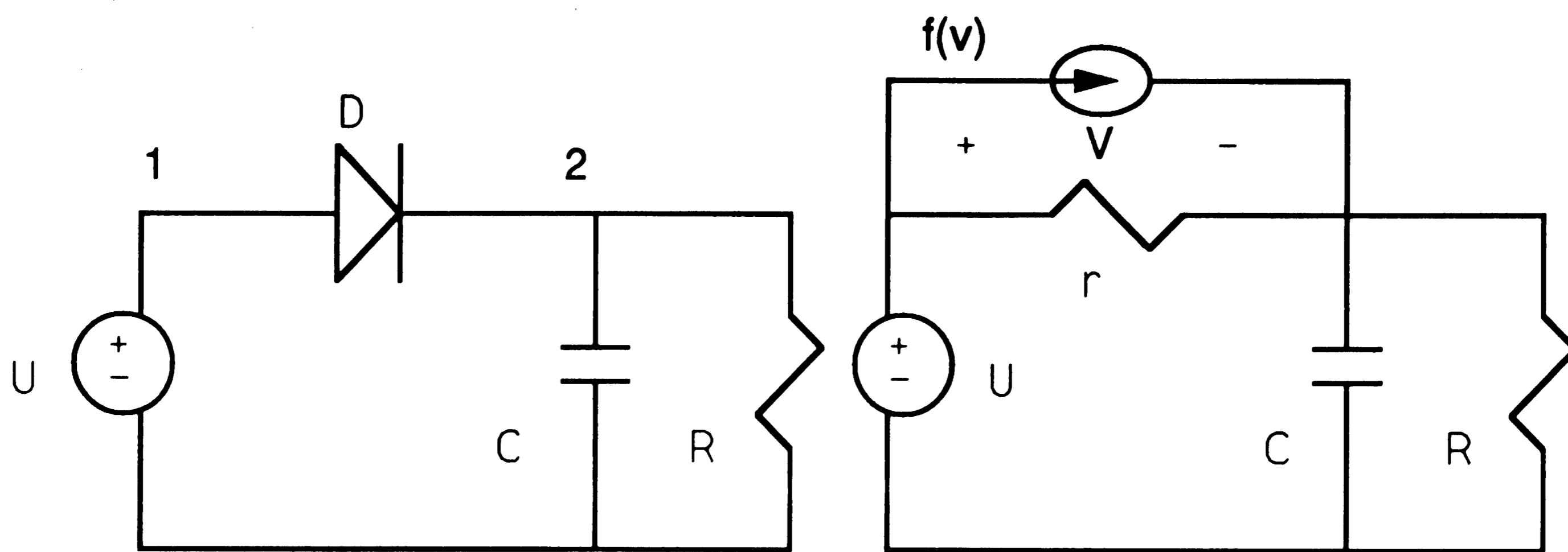


Figure 2.5a

Figure 2.5b

Figure 2.5

### A simple nonlinear circuit and its associated circuit

Note voltage equation :

$$\begin{vmatrix} 1 & 0 \\ -1/r & 1/r + 1/R + SC \end{vmatrix} \begin{vmatrix} V1 \\ V2 \end{vmatrix} = \begin{vmatrix} U \\ f(v) \end{vmatrix}$$

We can write

$$\begin{vmatrix} 1 & 0 \\ -1/r & 1/r + 1/R + SC \end{vmatrix} \begin{vmatrix} V1 \\ V2 \end{vmatrix} = \begin{vmatrix} 1 \\ 0 \end{vmatrix} U + \begin{vmatrix} 0 \\ 1 \end{vmatrix} f(v)$$

This is a form of equation (2.1)  $Y.Vn = a1.U + a2.f(v)$

And

$$\begin{vmatrix} V1 \\ V2 \end{vmatrix} = Y^{-1} \cdot \begin{vmatrix} 1 \\ 0 \end{vmatrix} U + Y^{-1} \begin{vmatrix} 0 \\ 1 \end{vmatrix} f(v) \quad (2.12)$$

$$v = Mv.Vn = \begin{vmatrix} 1 & -1 \end{vmatrix} \begin{vmatrix} V1 \\ V2 \end{vmatrix}$$

$$v = \left[1 - \frac{1}{r(1/r+1/R+SC)}\right]U - \frac{1}{(1/r+1/R+SC)}F(v)$$

This is of the form  $v = Au + BF(v)$

Note that

$$\frac{1}{r(1/r+1/R+SC)} = H(S)$$

Where  $H(S)$  is the transfer function of the circuit if we depress the nonlinear function  $f(v)$

We can write:

$$v(s) = [1-H(S)]U(S) - rH(S)F[v(S)]$$

$$\text{Denote } 1 - H(S) = G(s)$$

$$v(S) = G(S)U(S) - rH(S)F[v(S)]$$

$$\text{And } v(n) = g(n)*u(n) - rh(n)*f[v(n)]$$

$$v(n) = \sum_{k=0}^n g(n-k)u(k) - r \sum_{k=0}^{n-1} h(n-k)f[v(k)] - rh(0)f[v(n)]$$

Denote

$$\sum_{k=0}^n g(n-k)u(k) - r \sum_{k=0}^{n-1} h(n-k)f[v(k)] = w(n)$$

We have :

$$v(n) = w(n) - r.h(0).f[v(n)] \quad (2.13)$$

This is the form of equation (2.11) that we have derived above.

From (2.12) output  $v_o$ :

$$V_o = V_2 = \frac{1}{r(1/r+1/R+SC)}U + \frac{1}{(1/r+1/R+SC)}F(v)$$

Or

$$V_o = H(S)U(S) + rH(S)F[v(S)]$$

And

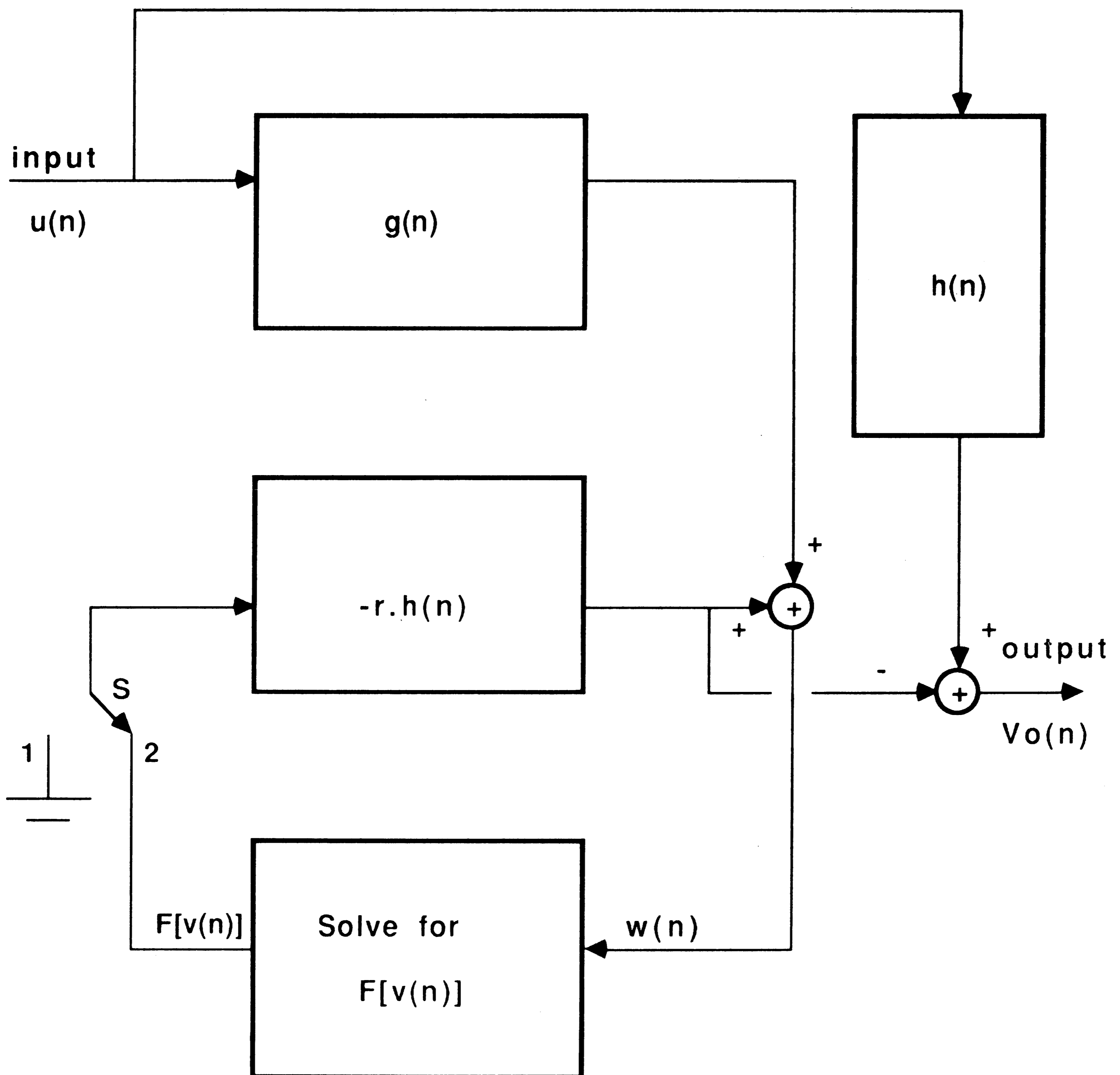


Figure 2.6a

Digital implementation of circuit in fig. 2.5a

$$v_o(n) = h(n)*u(n) + rh(n)*f[v(n)] \quad (2.14)$$

From (2.13) and (2.14) , the digital implementation of the original nonlinear analog circuit is depicted in figure 2.6a.

At index  $n$  switch  $S$  is at position 1 ,we have  $w(n)=u(n)*g(n)-rh(n)*F^{\wedge}[v(n)]$ . The solve-for  $-F[v(n)]$  circuitd receives this signal  $w(n)$  and generate  $F[v(n)]$ ,  $F[v(n)]$  is fed to the input of  $rh(n)$  circuit through the contact at position 2 of switch  $S$ . This new  $F[v(n)]$  updates the memory of digital filter  $rh(n)$  which in turn updates the output signal  $V_o(n)$ .

We have

$$H(S) = \frac{1}{r(1/R + 1/r + SC)}$$

Denote

$$\frac{1}{R} + \frac{1}{r} = \frac{1}{R'}$$

$$H(S) = \frac{1}{r(1/R' + SC)}$$

Using bilinear mapping

$$S = \frac{2}{T} \frac{1-Z^{-1}}{1+Z^{-1}}$$

to map  $H(S)$  into  $H(Z)$ , we obtain:

$$H(Z) = \frac{1}{r(1/R' + 2C(1-Z^{-1})/T(1+Z^{-1}))}$$

We can write  $H(Z)$  under the form:

$$H(Z) = \frac{a(1+Z^{-1})}{1-bZ^{-1}}$$

Where:

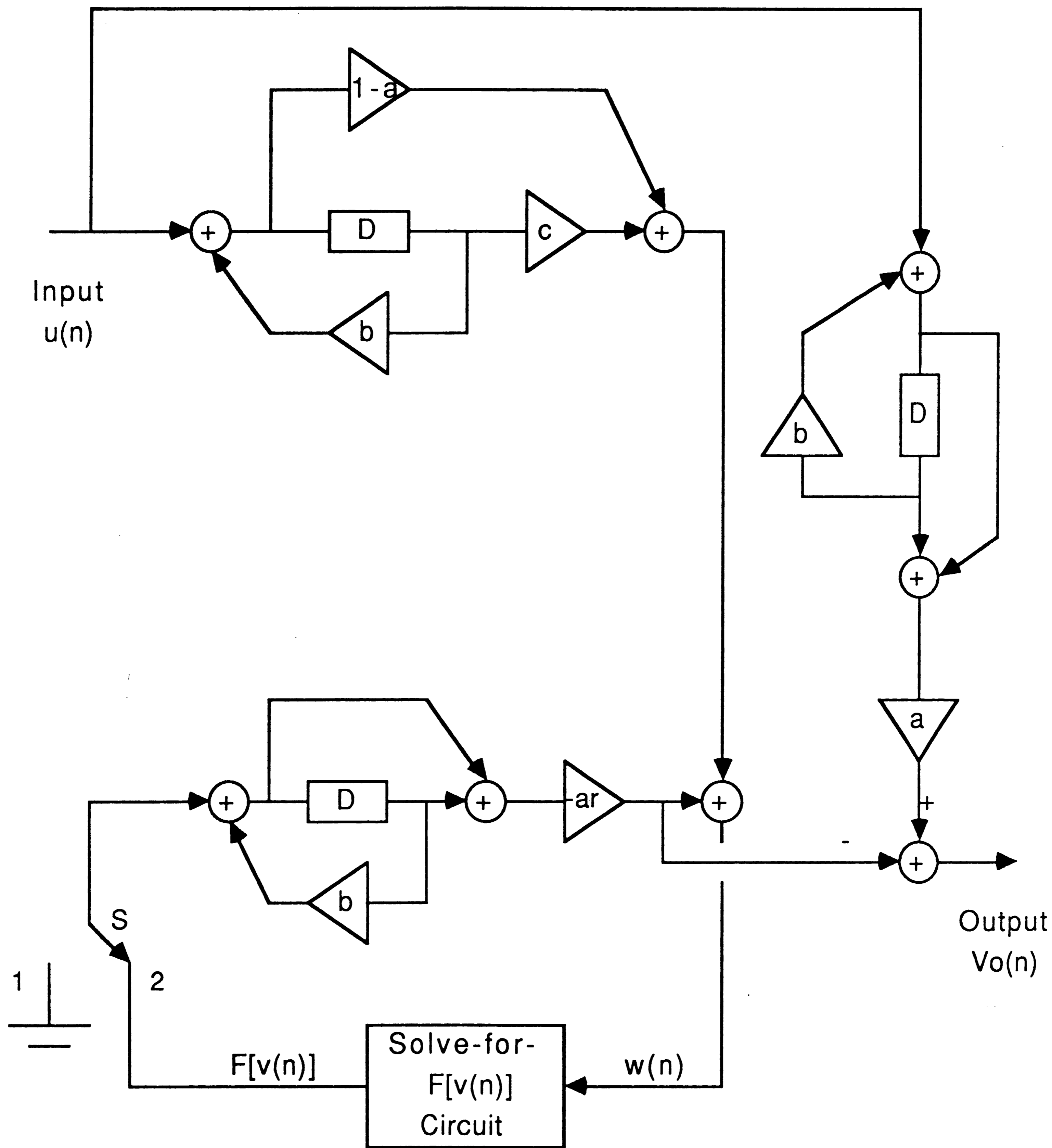


Figure 2.6b

Digital equivalent network of circuit in fig. 2.5a

$$a = \frac{R}{R+r+(2CRr)/T}$$

$$; b = \frac{2CR'-T}{2CR'+T}$$

T is the sampling period.

$$G(S)=1-H(S); G(Z)=1-H(Z)$$

$$G(Z) = \frac{(1-a)+cZ^{-1}}{1-bZ^{-1}}$$

Where  $c=-(a+b)$ . The digital equivalent circuit of this simple example is shown in figure 2.6b.

In this chapter a new method to realize digital equivalents from nonlinear analog networks as well as a particular process on a simple circuit have been presented. The digital equivalent network contains a sub-network N which solves for  $F[v(n)]$  according to equation (2.11).

In practical network, the matrix  $b(0)$  in equation (2.11) is usually a negative matrix as we saw in equation (2.13) of the example circuit. Thus, in general, equation (2.11) can be written as :

$$\mathbf{v}(n) = \mathbf{w}(n) - \mathbf{BF}[v(n)] \quad (2.15)$$

Providing that B is a non-negative matrix.

In the following chapter, the subcircuit to solve for  $F(v)$  associated with equation (2.15) are explored. Several circuit topologies and designs will be proposed for different nonlinear characteristics.

# Chapter 3

## Circuit topologies to realize equation

### $\mathbf{x} = \mathbf{w} - \mathbf{BF}(\mathbf{x})$ incorporated with DSP network

Chapter 2 has shown that for a given nonlinear network comprises of linear devices and two - terminal nonlinear devices together with independent sources, we can replace all nonlinearities by suitable models and obtain a standard equation has the form :

$$\mathbf{x} = \mathbf{A}\mathbf{u} + \mathbf{BF}(\mathbf{x}) \quad (3.1)$$

Where A and B are linear operators depend only on network topology, vector u stands for network inputs, vector x represents branch voltages and link currents associated with nonlinear models, and F(X) consists of dependent sources generated from the models.

The task of finding a digital equivalent from the given nonlinear network involves in finding the solution of equation:

$$\mathbf{x}(\mathbf{n}) = \mathbf{w}(\mathbf{n}) - \mathbf{BF}[\mathbf{x}(\mathbf{n})] \quad (3.2)$$

Where w(n) is an internal signal vector determined by network topology as well as the past values of F[x(n)] up to index (n-1)

$$w(n) = \sum_{k=0}^n a(n-k)u(k) + \sum_{k=0}^{n-1} b(n-k)F[x(k)] \quad (3.3)$$

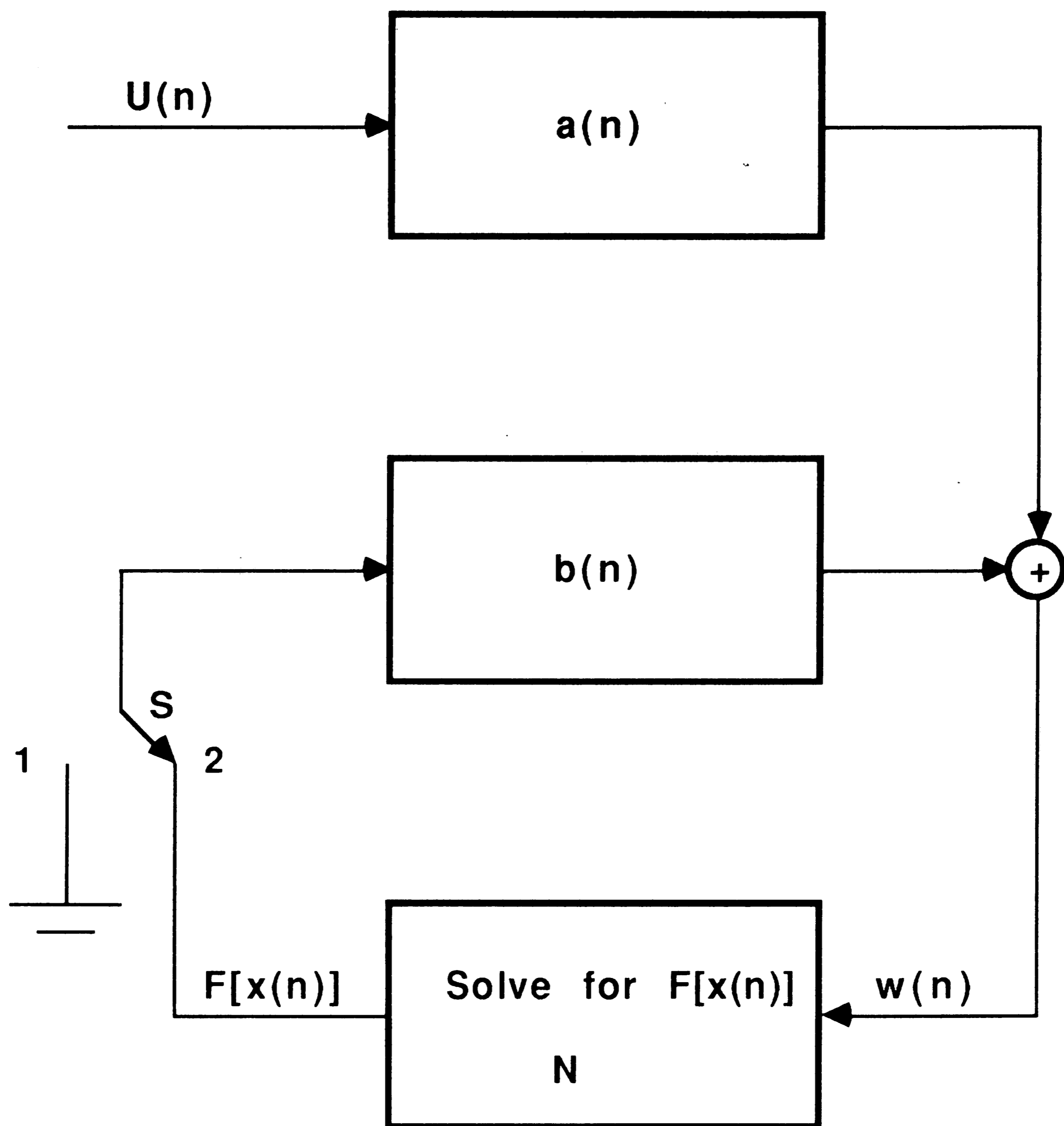


Figure 3.1

Simplified equivalent DSP network



### 3.1 A HARDWARE APPROACH

Figure (3.1) shows how equation (3.2) works in the associated DSP network.

The subcircuit N receives input  $w(n)$  and generates output  $F[x(n)]$  according to equation (3.2). Finding such a solution can be achieved by a software approach which utilizes some numerical method, such as Newton - Raphson, or Lin - Bairston method and their variations... The software solution, however, is not always convergent and more importantly, it's usually not fast enough to be implemented in real time.

The solution therefore is preferred to a hardware approach with the cost of some extra circuitries added to the DSP network as shown in figure 3.2

We have now:

$$\mathbf{x}(t) = \mathbf{w}(t) - \mathbf{BF}[\mathbf{x}(t)] \quad (3.4)$$

In figure 3.2 DAC and ADC are digital-to-analog and analog-to-digital converters.  $w(n)$  is converted to analog form by the DAC. The solve-for- $f[x]$  circuit now operates in continuous time domain. The analog output  $f[x(t)]$  of the solve-for- $F[x]$  circuit is then converted back to digital form by the ADC. We will see later that the DAC is a part of Solve-For- $F[x(t)]$  circuit if they are well designed; and therefore the fact that a DAC is added to the network is not a serious problem, and the practical network of figure 3.2 can be more simplified.

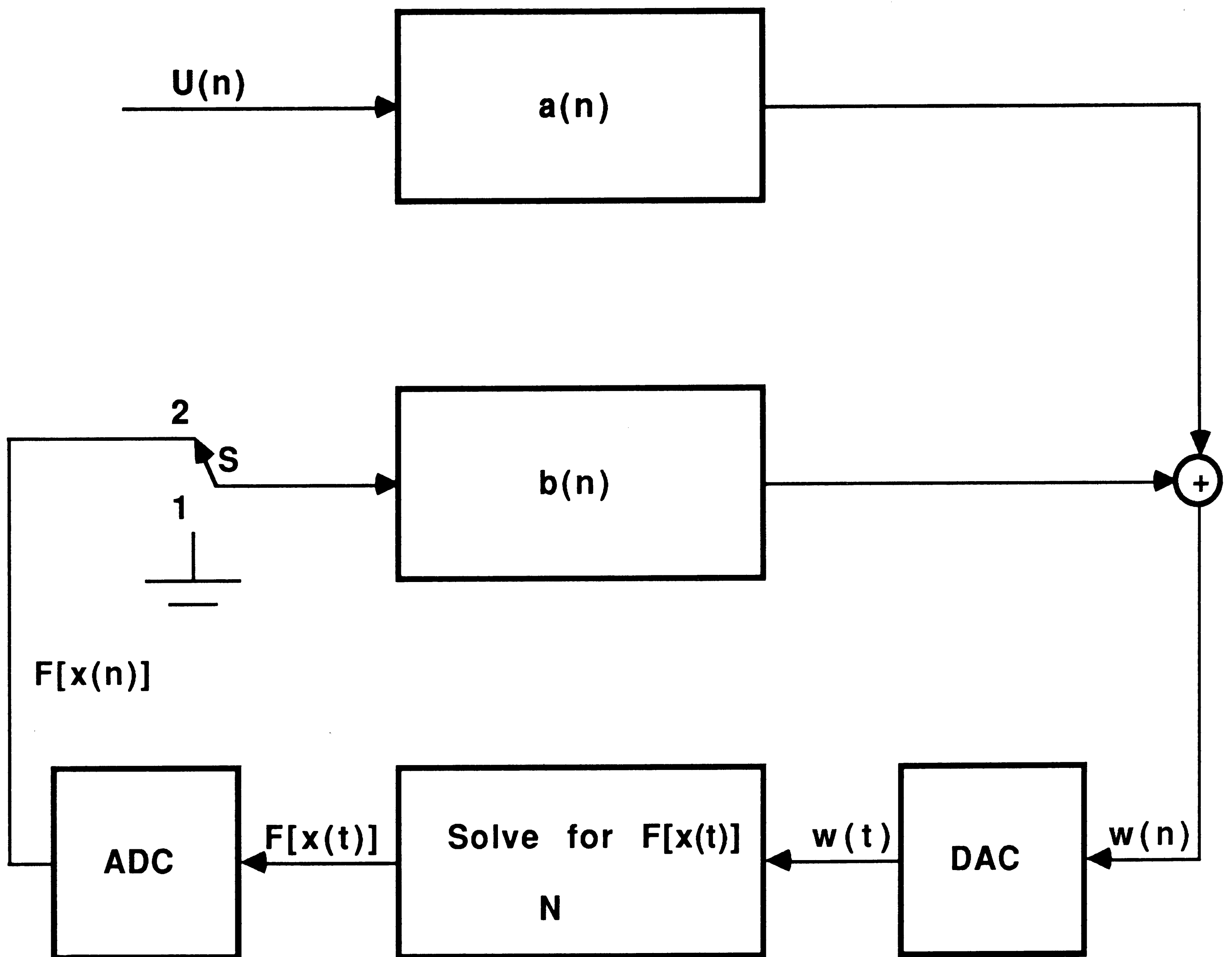


Figure 3.2

A hardware approach to solve for  $F[x(n)]$

## 3.2 FIRST ORDER EQUATION

### 3.2.1 Circuit topology

We consider first the case when the given network contains only one non-linearity.

In this circumstance, equation (3.4) contains only constants, variable and nonlinear function in  $R^1$ .

If we agree that  $F[x]$  is a voltage-controlled current source with nonlinear characteristics and that  $F[x]$  can be either a one-port or two-port network, then the circuit topologies in figure 3.4 will be a good candidate for realization of equation (3.4)

In figure (3.4)  $D\{w(n)\}$  : Binary code of  $w(n)$

And

$$I_A = f_A(V_{ref}, D\{w(n)\})$$

With  $f_A$  is a function defined by DAC. From the circuit of figure 3.4, either a or b, we can derive easily:

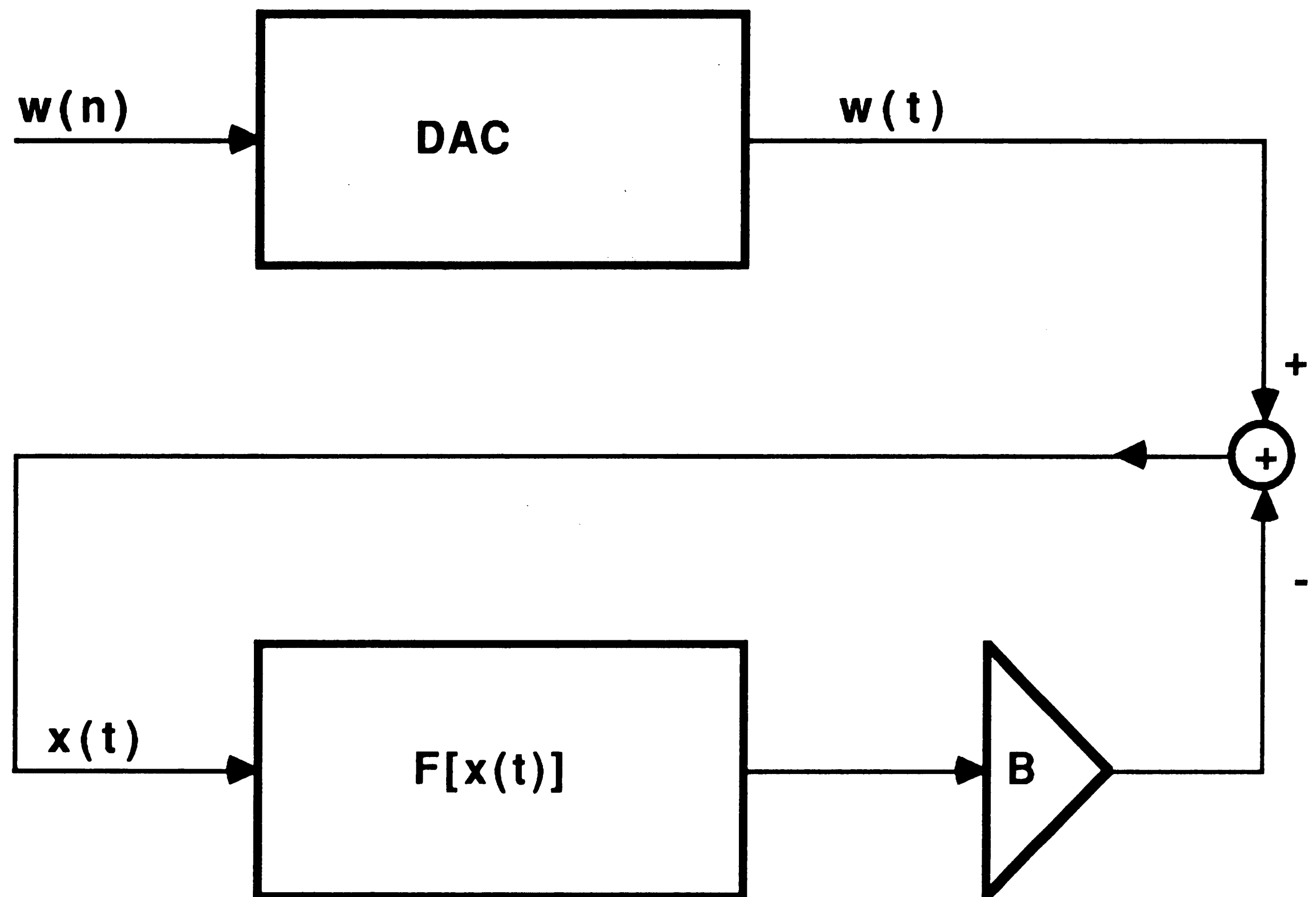
$$\begin{aligned} x &= BI_A - BF[x] \\ x &= Bf_A(V_{ref}, D\{w(n)\}) - BF[x] \end{aligned} \quad (3.5)$$

In which we can design the D/A converter such that:

$$Bf_A(V_{ref}, D\{w(n)\}) = w(t) \quad (3.6)$$

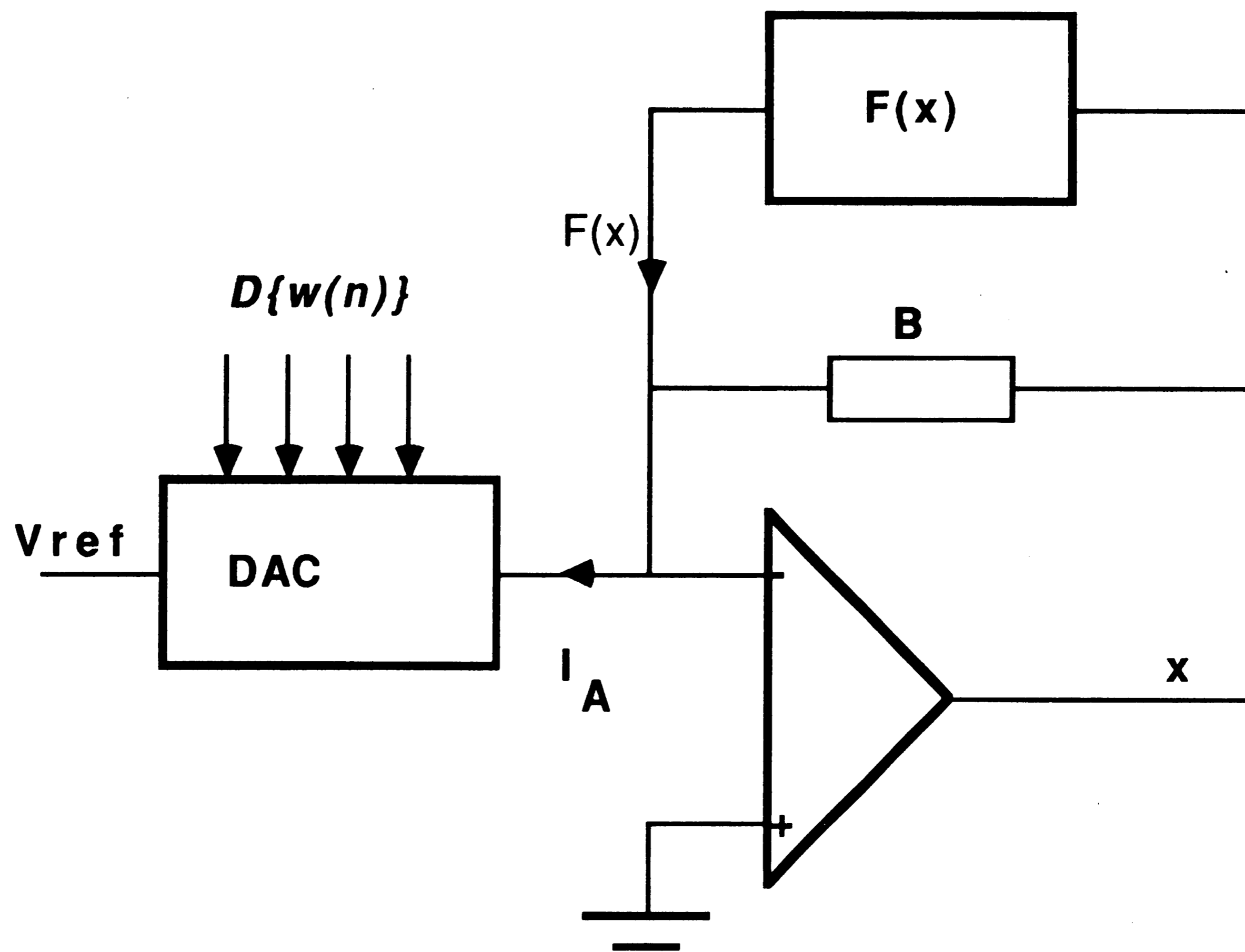
Equation (3.6) indicates that we can adjust  $w(t)$  depending on particular given nonlinear network.

The settling time of a DAC can be on the order of a fraction of a micro-second. Op-amp gain -bandwidth product will have a large impact on the settling time of the entire circuit. With a high speed Op-amp the over all settling time can be achieved in a range of few

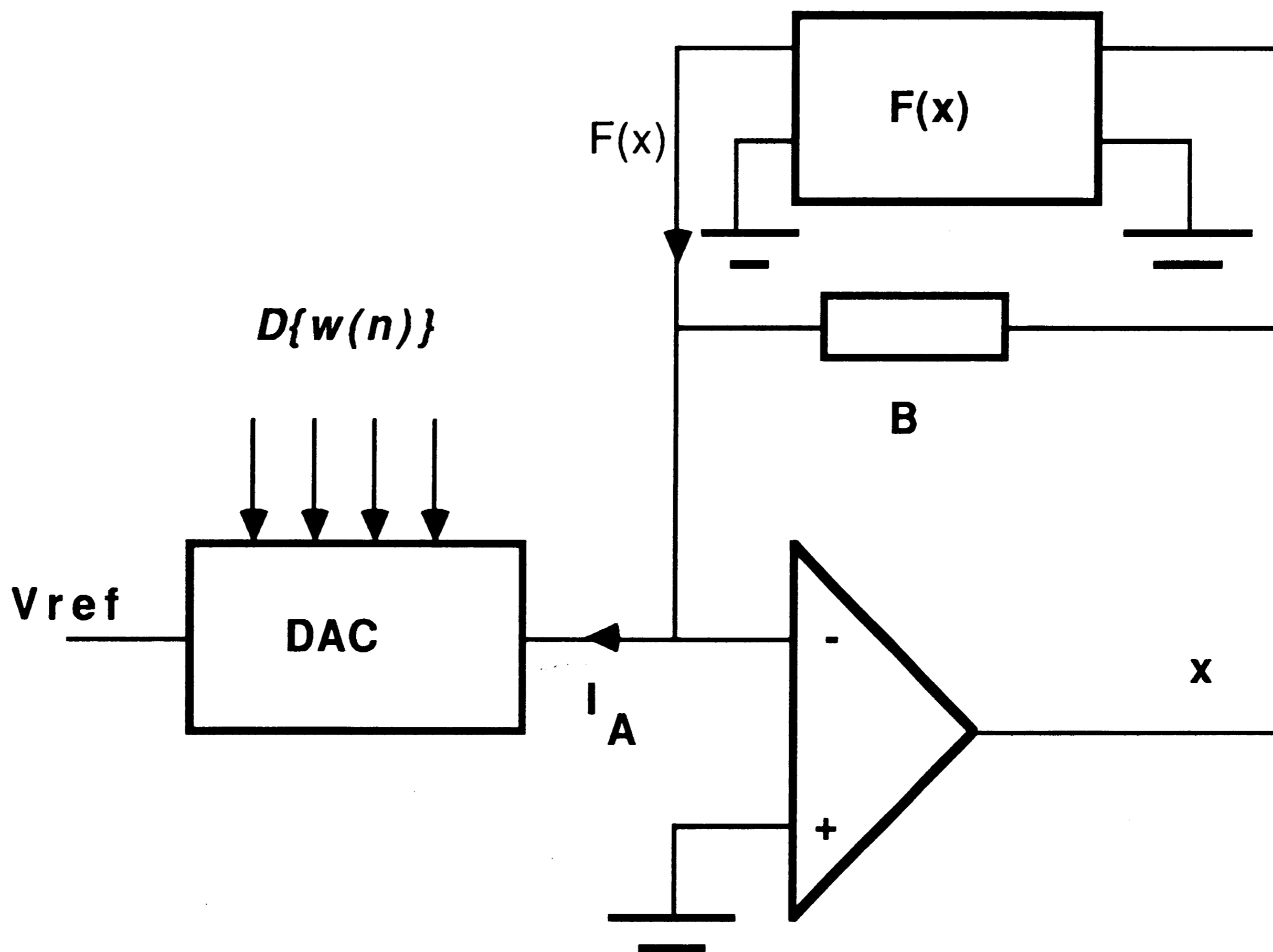


**Figure 3.3**

**A model for equation (3.4)**



a. one-port  $F(x)$



b. Two-port  $F(x)$

Figure 3.4 Circuit realization of equation (3.4)

hundreds of nano seconds.

### 3.2.2 Programming gain B

In equation (3.4) , depends on particular nonlinear element , we may desire to change the gain B. A convenient way is to replace the fixed impedance B in figure 3.4 by a multiplying DAC with an R-2R ladder in figure 3.5:

Due to the virtual ground at inverting node of Op-amp, the voltages at node A, B, C, D are constant regardless of switch positions.

$$V_D = 2^{-1}V_C = 2^{-2}V_B = 2^{-3}V_A = 2^{-3}V$$

And:

$$I_4 = 2^{-1}I_3 = 2^{-2}I_2 = 2^{-3}I_1 = 2^{-3}V/2R$$

$$I_O = b_1I_1 + b_2I_2 + b_3I_3 + b_4I_4$$

Where  $b_i$ 's are digital bits applied to switches.

$$I_O = V/R(b_12^{-1} + b_22^{-2} + b_32^{-3} + b_42^{-4})$$

Denote:

$$D = \sum_{i=1}^4 b_i 2^{-i}$$

We have:

$$I_o = DV/R$$

In general , for a multiplying DAC configuration of figure 3.5 the current  $I_o$  is a function of input voltage V, ladder resistor R and digital code D.

$$I_o = f(V,R,D) = DV/R \quad (3.7)$$

Now let consider the circuit of figure 3.6. In figure 3.6 both DAC 1

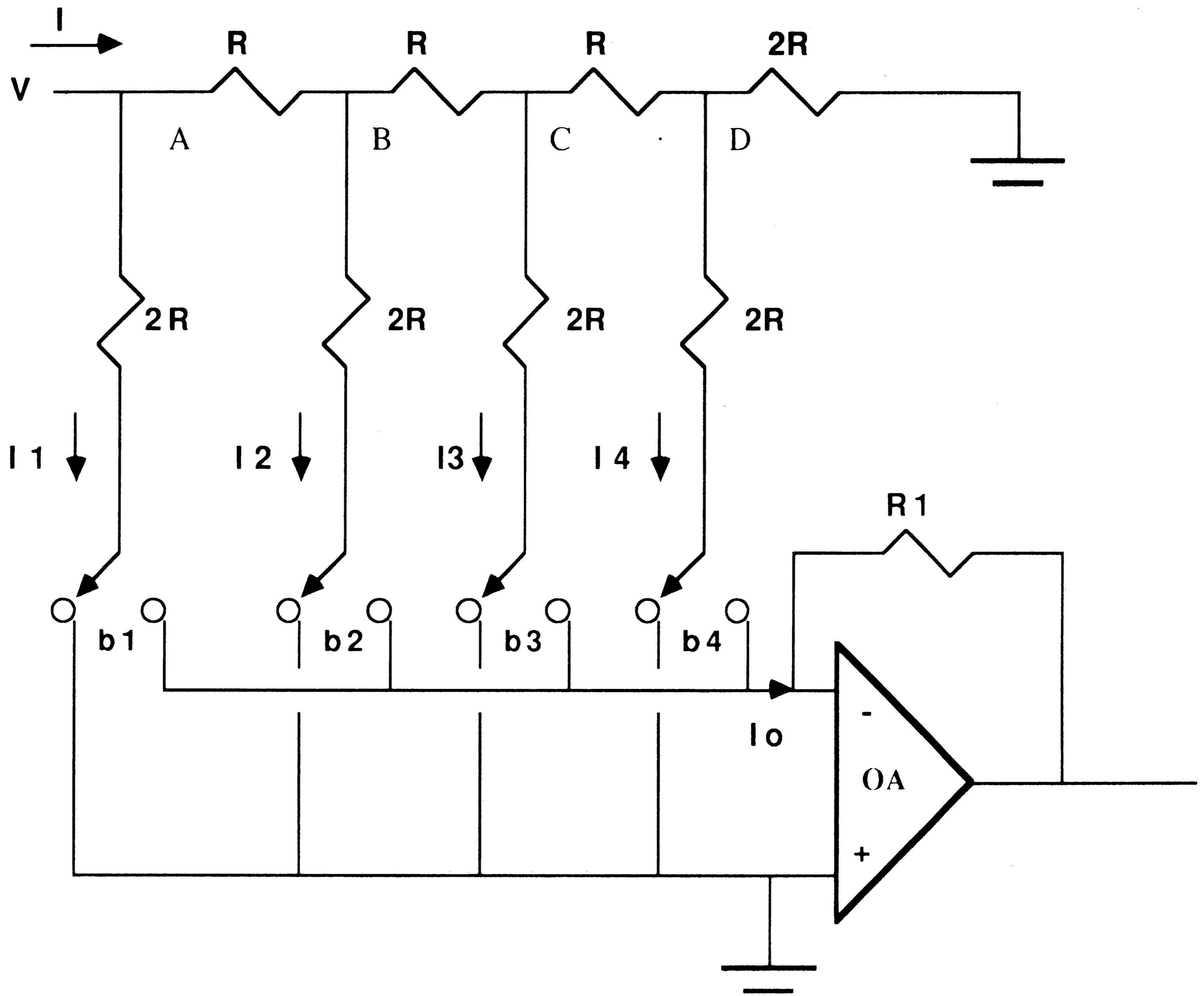


Figure 3.5

Current-mode R-2R ladder 4 bit Multiplier

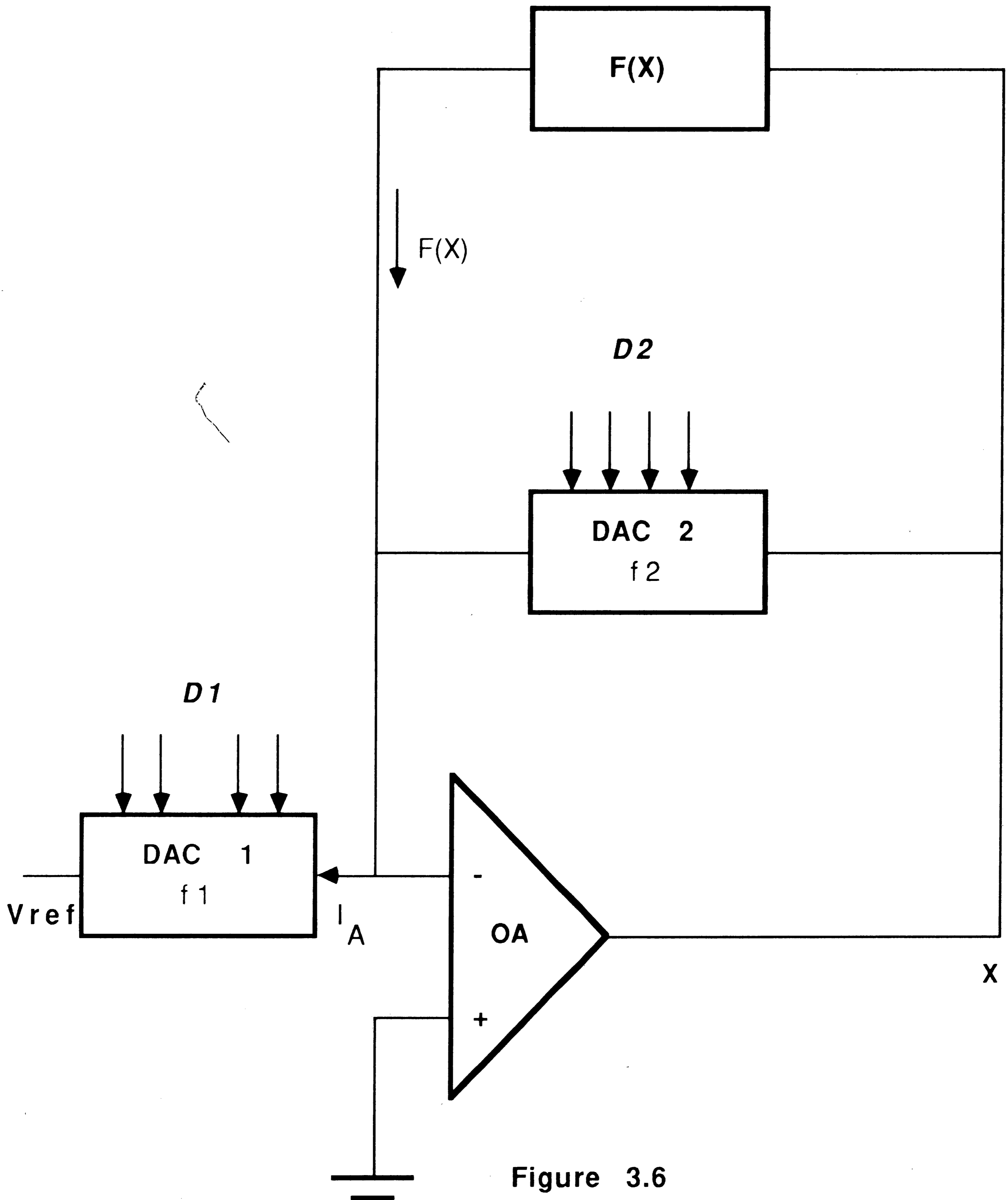


Figure 3.6  
Programming gain B for one-port  $f(x)$



and ADC 2 are multiplying DAC type of figure 3.5, with the resistor ladders  $R_1-2R_1$  and  $R_2-2R_2$  respectively. D1 is the binary code of input  $w(n)$ , D2 is the programming binary code, used to program the gain B

We can analyze circuit of figure 3.6 easily by superposition principle:

1. Circuit without current source  $F(x)$

$$\text{From DAC 1 } I_A = -f_1(V_{ref}, R_1, D1) = -D1V_{ref}/R_1$$

$$\text{From DAC 2 } I_A = f_2(x_1, R_2, D2) = D2x_1/R_2$$

$$x_1 = -V_{ref} \frac{R_2 D1}{R_1 D2} \quad (3.8)$$

2. Circuit without current source  $f1$

We have:

$$F[x] = -f_2(x_2, R_2, D2) = -D2x_2/R_2$$

$$x_2 = -F[x]R_2/D2 \quad (3.9)$$

From (3.8) and (3.9)

$$x = x_1 + x_2 \quad (3.10)$$

$$x = -V_{ref} \frac{R_2 D1}{R_1 D2} - F[x] \frac{R_2}{D2}$$

Equation (3.10) explicitly shows a way to program the gain B by programming the binary code D2.

Note that DAC 1 and DAC 2 can use the available op-amp OA in figure 3.6 as a part of their circuits. Therefore actually DAC 1 and DAC2 contain only resistor ladder and transistor switches, this fact makes the circuit in figure 3.6 becomes more simpler than it appears.

One drawback of circuit in figure 3.6 is the gain B is reversely proportional to D2. When nonlinear function  $F[x]$  is generated by a two-port network, We can modify the circuit in figure 3.6 to avoid this problem . A modification is shown in figure 3.7 for two-port nonlinear  $F[x]$

In the circuit of figure 3.5,  $V = RI$ , replace this in equation (3.7) we get

$$I_o = ID \quad (3.11)$$

Equation (3.11) indicates that the multiplying DAC type of figure 3.5 can be used as a programmable current amplifier (PCA). In figure 3.7 DAC 2 controls output current  $F[x]$  by means of programming binary code D2.

We have from figure 3.7:

$$\mathbf{x} = -V_{ref} D1 R_B / R_1 - R_B D2 F[\mathbf{x}] \quad (3.12)$$

In this case the gain B is directly proportional to binary code D2

### 3.3 SECOND AND HIGHER ORDER EQUATIONS

#### 3.3.1 Second order equation

When The given network has two nonlinearities characterized by  $F_1[x_1]$  and  $F_2[x_2]$  , the standard form equation (3.4) gives a second order equation:

$$x_1 = w_1 - B_{11}F_1(x_1) - B_{12}F_2(x_2)$$

$$x_2 = w_2 - B_{21}F_1(x_1) - B_{22}F_2(x_2)$$

A model for this second order equation is shown in figure 3.8

When all nonlinearities are simulated by two-port networks,

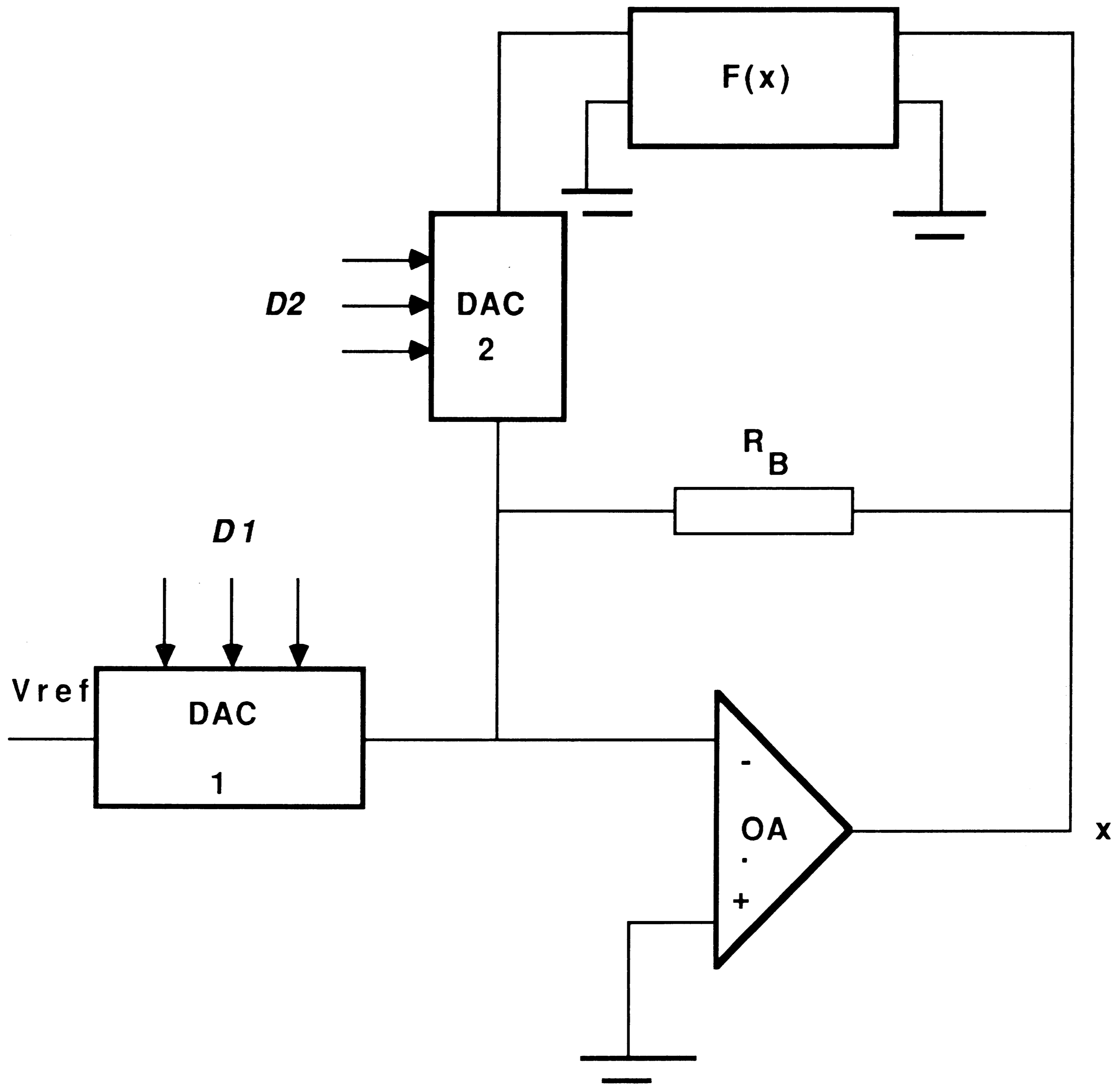


Figure 3.7

Programming gain B for two-port  $F(x)$

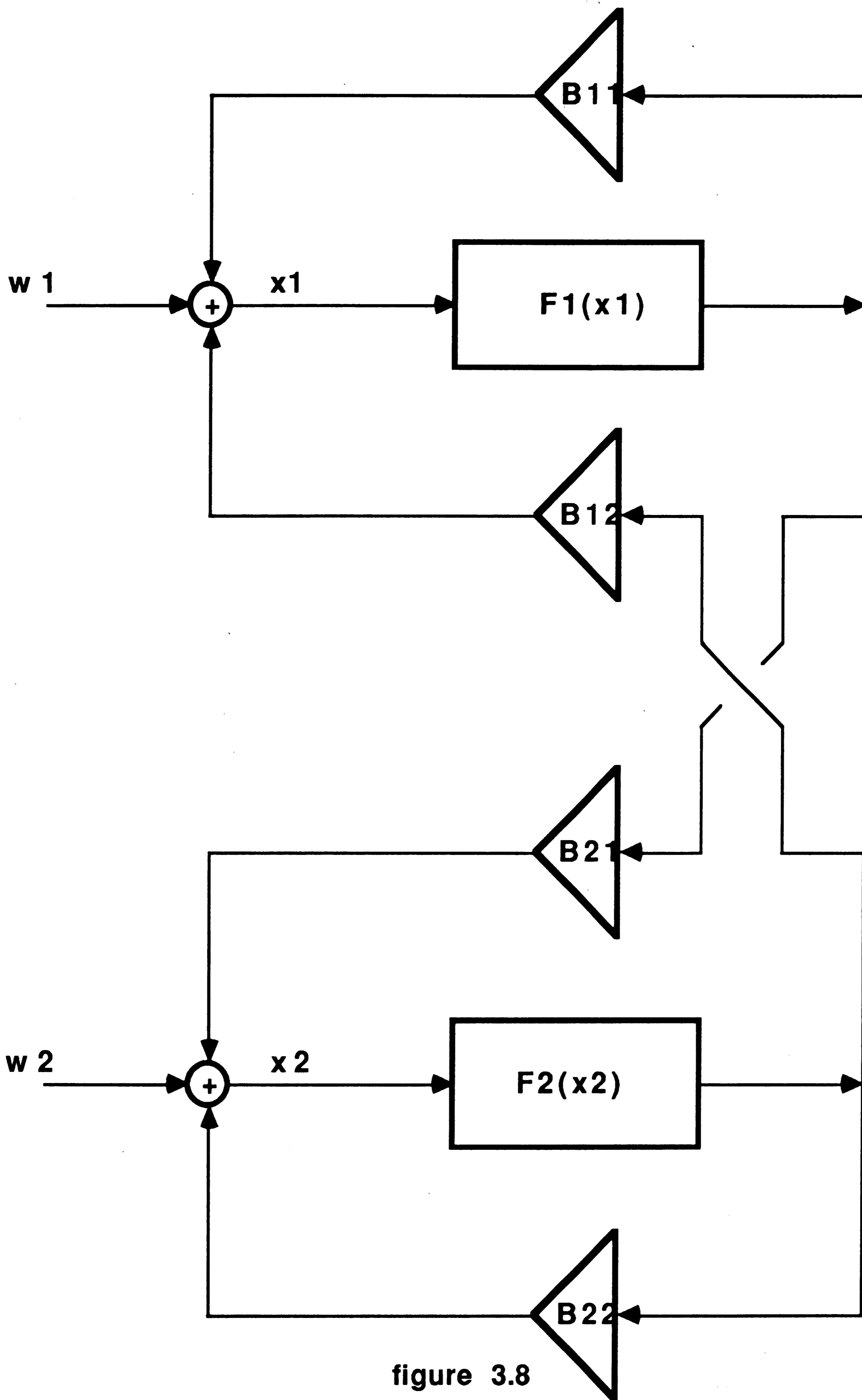


figure 3.8

A model for second order equation

it's straight forward to get a circuit configuration to realize this second order equation. Such a circuit configuration is shown in figure 3.9

When function  $F_1(x_1)$  and  $F_2(x_2)$  are generated by one-port network, we need some modification from the circuit of figure 3.9. Let consider the circuit of figure 3.10

In figure 3.10, the output current  $I_A$  of digital-to-analog converter:

$$I_A = f(V_{ref}, D)$$

At output of op-amp

$$V_{o1} = B_1 f(V_{ref}, D) - B_1 F_1[x_1]$$

We also have

$$V_{o1} = x_1 + B_2 F_1[x_1]$$

Therefore:

$$x_1 + B_2 F_1[x_1] = B_1 f(V_{ref}, D) - B_1 F_1[x_1]$$

$$x_1 = B_1 f(V_{ref}, D) - (B_1 + B_2) F_1[x_1]$$

If now we make  $B_1 + B_2 = B_{11}$  then

$$x_1 = B_1 f(V_{ref}, D) - B_{11} F_1[x_1] \quad (3.13)$$

$$V_{o2} = B_2 F_1[x_1] \quad (3.14)$$

Equation (3.13) give the desired coefficient  $B_{11}$ . Voltage  $V_{o2}$  is fed to another circuit with similar configuration to provide the coefficient  $B_{21}$ . The entire modified configuration for one-port nonlinear functions  $F[x]$ 's is shown in figure 3.11.

In figure 3.11 , it's straight forward to show that:

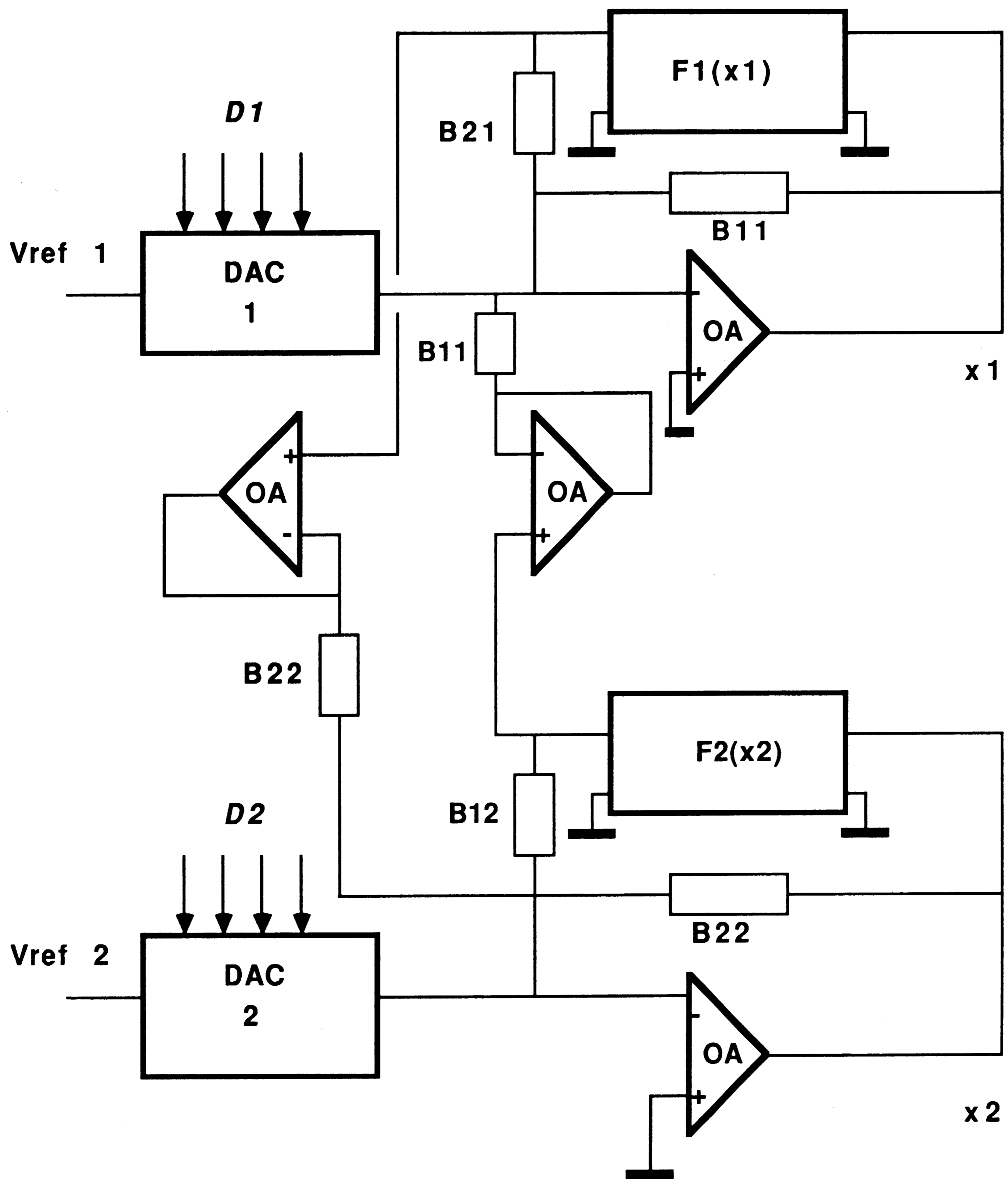


Figure 3.9

Circuit for 2nd order equation with two-port  $F(x)$

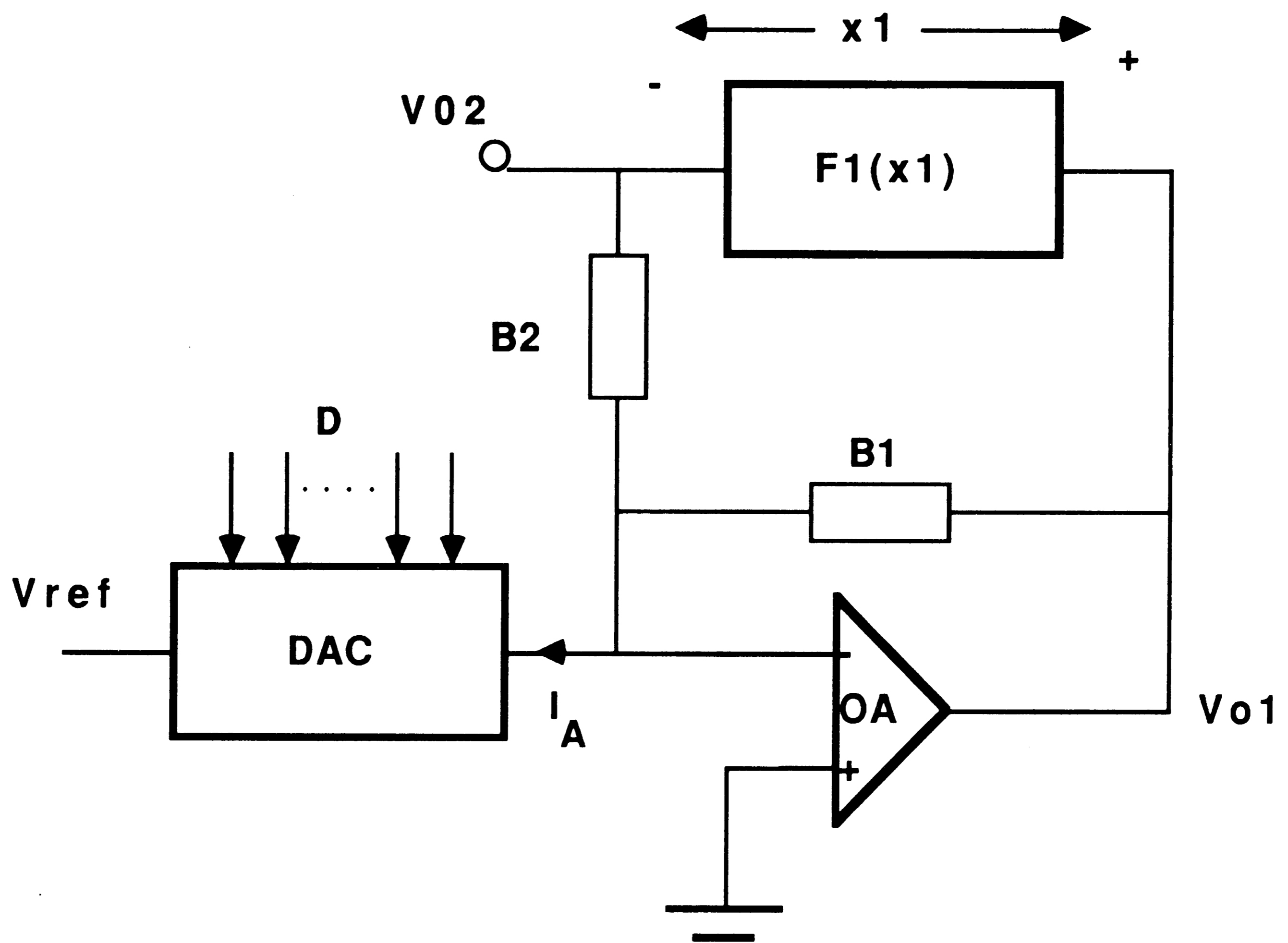


figure 3.10

Circuit configuration for one-port  $F(x)$

$$\mathbf{B}_{11} = \mathbf{B}_1 + \mathbf{B}'_{21} \quad (3.15)$$

$$\mathbf{B}_{12} = \mathbf{B}'_{12} \frac{\mathbf{B}_1}{\mathbf{B}'_1} \quad (3.16)$$

$$\mathbf{B}_{21} = \mathbf{B}'_{21} \frac{\mathbf{B}_2}{\mathbf{B}'_2} \quad (3.17)$$

$$\mathbf{B}_{22} = \mathbf{B}_2 + \mathbf{B}'_{12} \quad (3.18)$$

And

$$x_1 = w_1 - B_{11}F_1(x_1) - B_{12}F_2(x_2)$$

$$x_2 = w_2 - B_{21}F_1(x_1) - B_{22}F_2(x_2)$$

Note that the voltages  $x_1$  and  $x_2$  are not at Op-amp outputs, but this causes no problem because in the associated DSP network what we need are  $F_1[x_1]$  and  $F_2[x_2]$  rather than  $x_1$  and  $x_2$ .

To program the coefficients  $B_{ij}$ 's, we can use the circuits of multiplying DAC type to replace fixed resistors  $B_{ij}$  as in the case of first order equation.

### 3.3.2 Higher order equations

Thanks to the simplicity in circuit topology, the network of either figure 3.9 or figure 3.11 can be generalized for higher order equations. Figure 3.12 shows a network for 3rd order equation.

$$x_1 = w_1 - B_{11}F_1(x_1) - B_{12}F_2(x_2) - B_{13}F_3(x_3)$$

$$x_2 = w_2 - B_{21}F_1(x_1) - B_{22}F_2(x_2) - B_{23}F_3(x_3)$$

$$x_3 = w_3 - B_{31}F_1(x_1) - B_{32}F_2(x_2) - B_{33}F_3(x_3)$$

In figure 3.12,  $B_{ij}$ 's are designed according to following conditions:



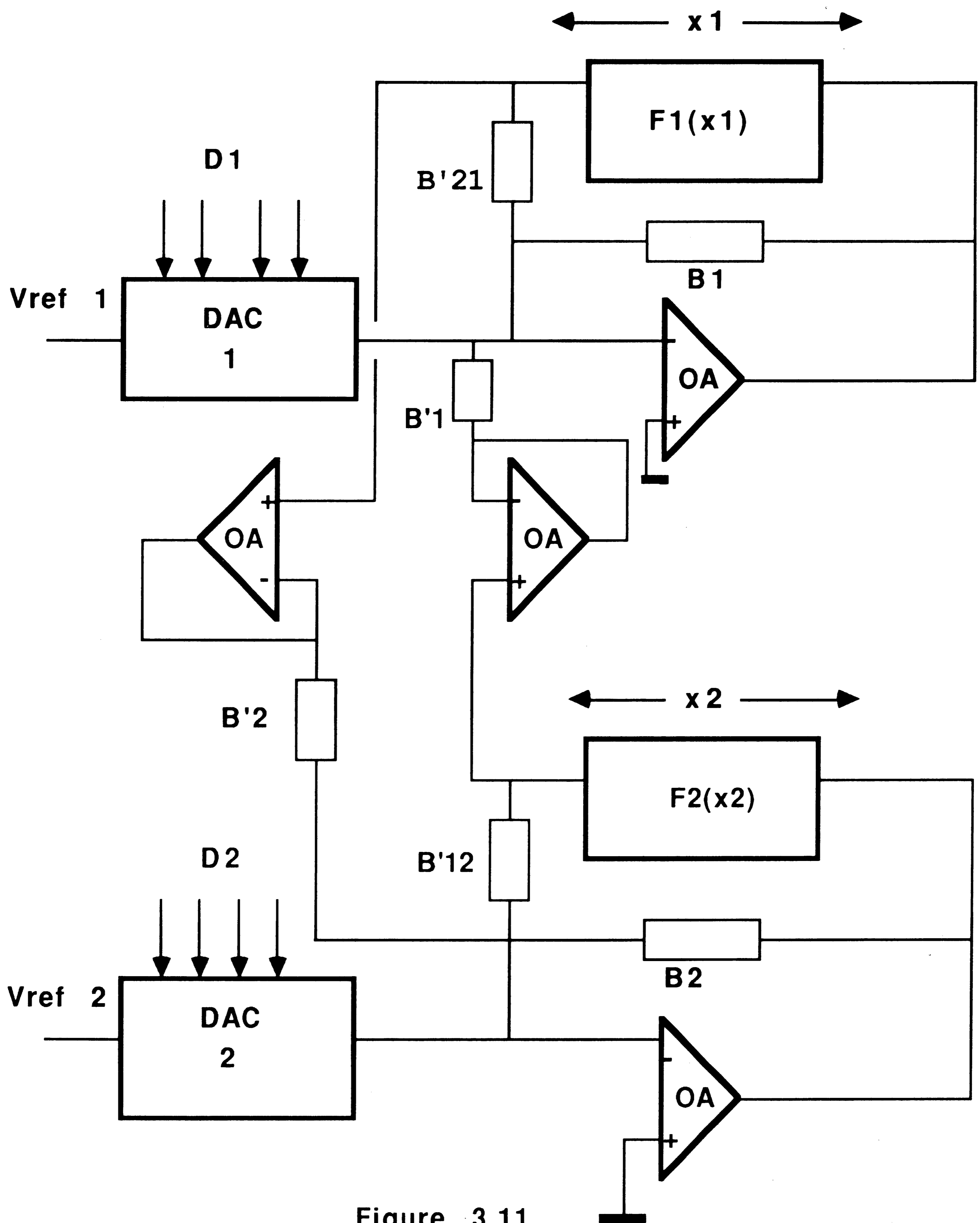


Figure 3.11

Circuit for 2nd order equation with one-port  $F(x)$

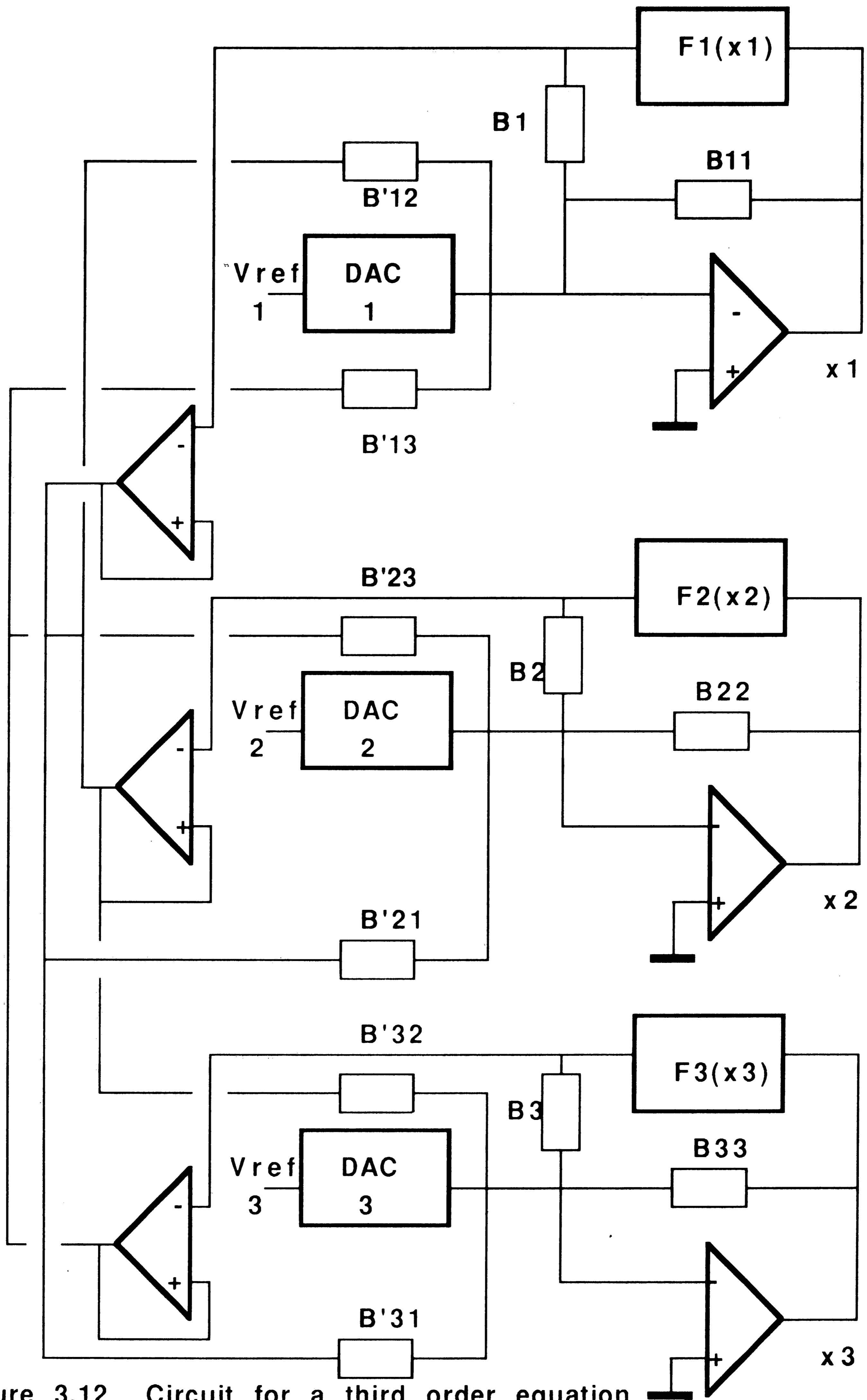


Figure 3.12 Circuit for a third order equation

$$\frac{B_{11}}{B'_{12}}B_2=B_{12}$$

$$\frac{B_{11}}{B'_{13}}B_3=B_{13}$$

$$\frac{B_{22}}{B'_{21}}B_1=B_{21}$$

$$\frac{B_{22}}{B'_{23}}B_3=B_{23}$$

$$\frac{B_{33}}{B'_{31}}B_1=B_{31}$$

$$\frac{B_{33}}{B'_{32}}B_2=B_{32}$$

In general, to create an  $n^{\text{th}}$  order subcircuit  $N$  to solve for  $F(\mathbf{x}) = [f_1(x_1), f_2(x_2), \dots, f_n(x_n)]^T$  we need  $2n$  op-amp's and  $n$  multiplying -type DAC which consists of only resistor ladders and transistor switches. Due to the low resistor ratio in R-2R ladders, an  $n^{\text{th}}$  order subcircuit  $N$  can be well fabricated by monolithic integrated circuit technology.

# Chapter 4

## Programmable Polynomial Module incorporated with nonlinear DSP network

In this chapter we design a polynomial module which will be used in association with nonlinear DSP network. The polynomial module generates function :

$$f(x) = \sum_{i=0}^n a_i x^i$$

Where coefficients  $a_i$ 's can be programmed by the user to make the module becomes versatile. Such a module can simulate arbitrary nonlinear characteristics because in some interest range of variable  $x$  any differentiable function can be represented by a polynomial form.

We restricted ourselves to design a module with voltage-controlled current source output rather than other forms. A current source output turned out to be the most convenient way when it was used with associated nonlinear DSP network as we already shown in chapter 3. With high speed and performance required for associated nonlinear DSP network in mind we preferred to bipolar technology. Bipolars provide fast current switching and true output current source or sink capability. When used in nonsaturating current-steering mode, bipolar transistors switch very rapidly, typically within nano-seconds. Moreover, with true current source or sink capability, the module output can be converted to a voltage merely by a resistor termination, thus avoiding the additional delays of an I-V

converter.

## 4.1 THE PRINCIPLE

The heart of our module is the transconductance multiplier of figure 4.1, consisting of differential pair  $Q_3$  and  $Q_4$  to provide variable transconductance. Since its conception was introduced [11] the linearized transconductance multiplier rapidly gained acceptance as the preferred approach to the realization of monolithic analog multipliers, and its simplicity has commended it for use in low-cost modular design. Accuracy of these units and drift and noise performance have been developed, future improvements in precision bipolar technology will almost certainly result in accuracies of 0.1 percent becoming commonplace.

In figure 4.1, diode D1 and D2 are used to provide the proper base driver for the differential pair  $Q_3$  and  $Q_4$ .

$$\begin{aligned} \text{By Kirchoff's voltage law, } V_{D1} + V_{BEQ4} &= V_{D2} + V_{BEQ3} \\ V_{D1} - V_{D2} &= V_{BEQ3} - V_{BEQ4} \end{aligned} \quad (4.1)$$

Assume negligible base currents :

$$I_1 = I_{S1}[e^{V_{D1}/V_T} - 1] \approx I_{S1}e^{V_{D1}/V_T}$$

And

$$V_{D1} = V_T \ln[I_1/I_{S1}]$$

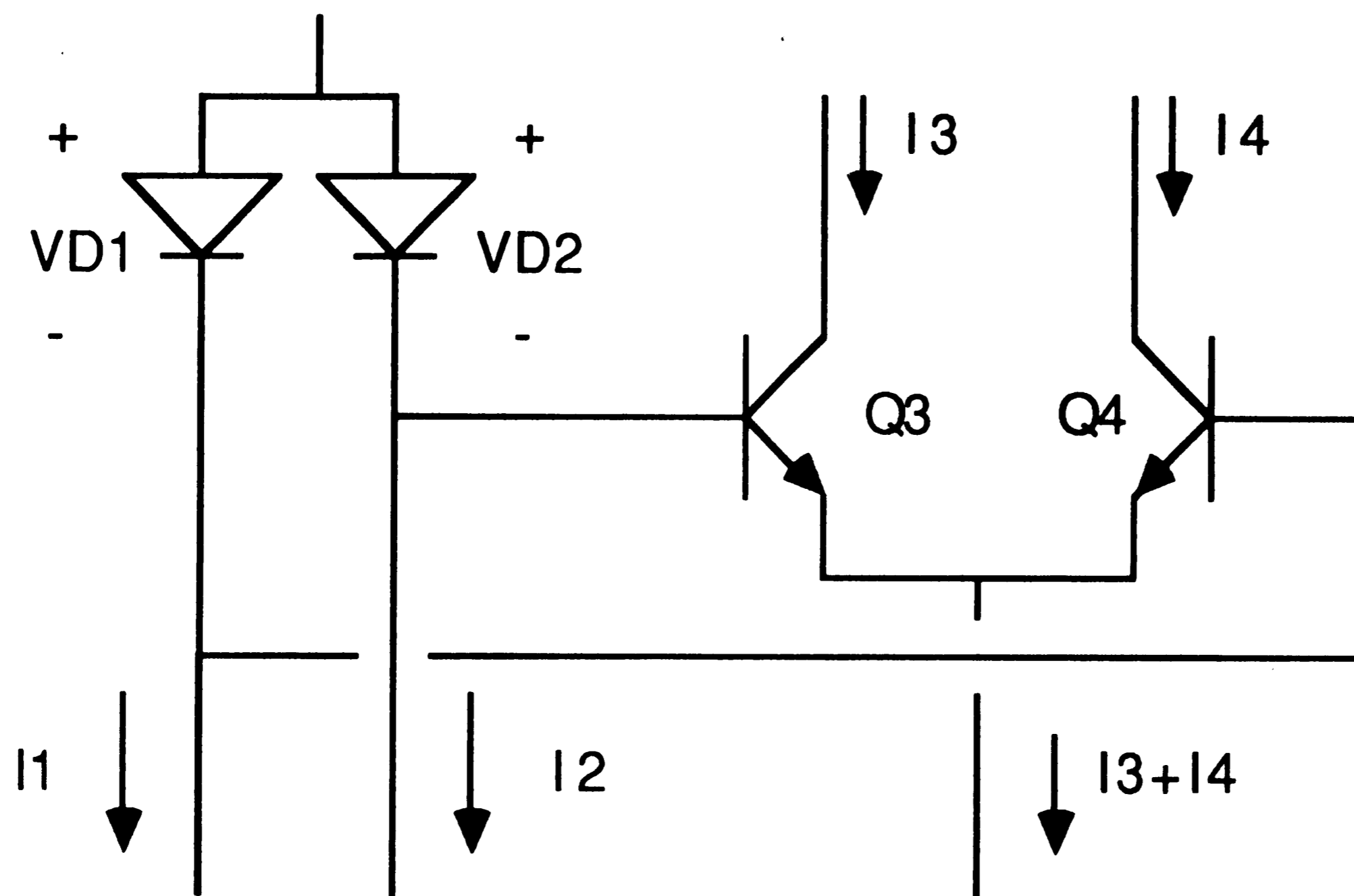
From (4.1)

$$V_T \ln[I_1/I_{S1}] - V_T \ln[I_2/I_{S2}] = V_T \ln[I_3/I_{S3}] - V_T \ln[I_4/I_{S4}]$$

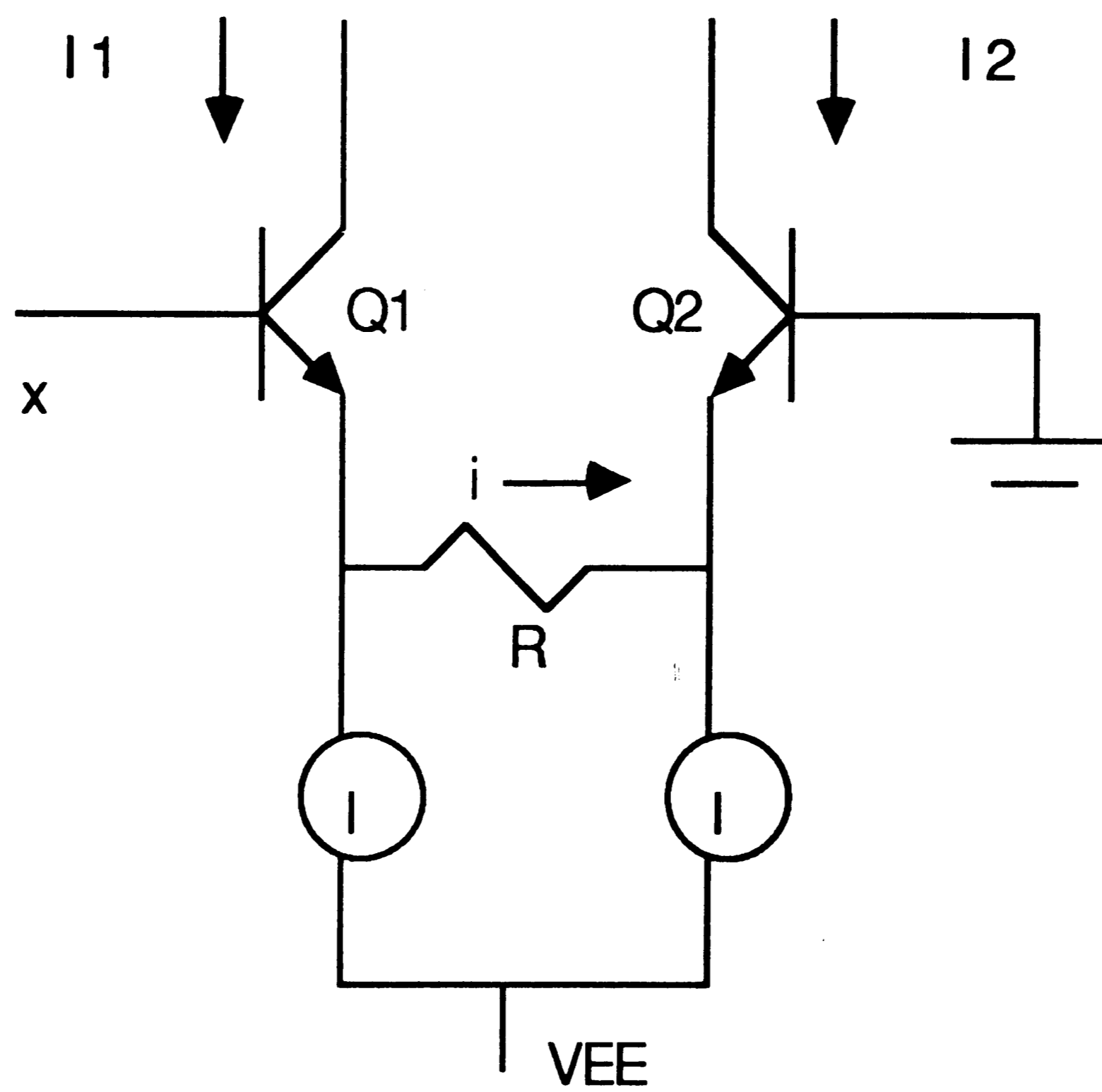
If the transistors are well matched, then

$$\ln[I_1/I_2] = \ln[I_3/I_4]$$

And:



**Figure 4.1**  
**Transconductance multiplier**



**Figure 4.2**  
**Differential V-I converter**

$$\frac{I_1 - I_2}{I_3 + I_4} = \frac{[I_1 - I_2][I_3 + I_4]}{[I_1 + I_2]} \quad (4.2)$$

Equation (4.2) indicates the circuit's ability to multiply different current ( $I_1 - I_2$ ) by total emitter current ( $I_3 + I_4$ ).

The differential current  $I_1 - I_2$  can be obtained from the emitter-degenerated amplifier in figure 4.2

In figure 4.2

$$I_1 = I + i$$

$$I_2 = I - i$$

$$i = 1/2(I_1 - I_2) \quad (4.3)$$

By Kirchoff's voltage law :

$$x = V_{BEQ1} + Ri - V_{BEQ2} \quad (4.4)$$

From (4.3) and (4.4) :

$$x = R/2(I_1 - I_2) + V_T \ln[I_1/I_2] \quad (4.5)$$

In a well- designed circuit the 2nd term of the right hand side in equation (4.5) is negligible compared to other terms, therefore to a first approximation

$$I_1 - I_2 = \frac{2}{R}x \quad (4.6)$$

Equation (4.6) already gives 1st order power of  $x$ ; all we need is converting differential output into single ended output. This is a straight forward task. For higher order power of  $x$ , equation (4.2) and (4.6) associate with circuits of figure (4.1) and (4.2) respectively can be arranged in some way as we will see in the following section.

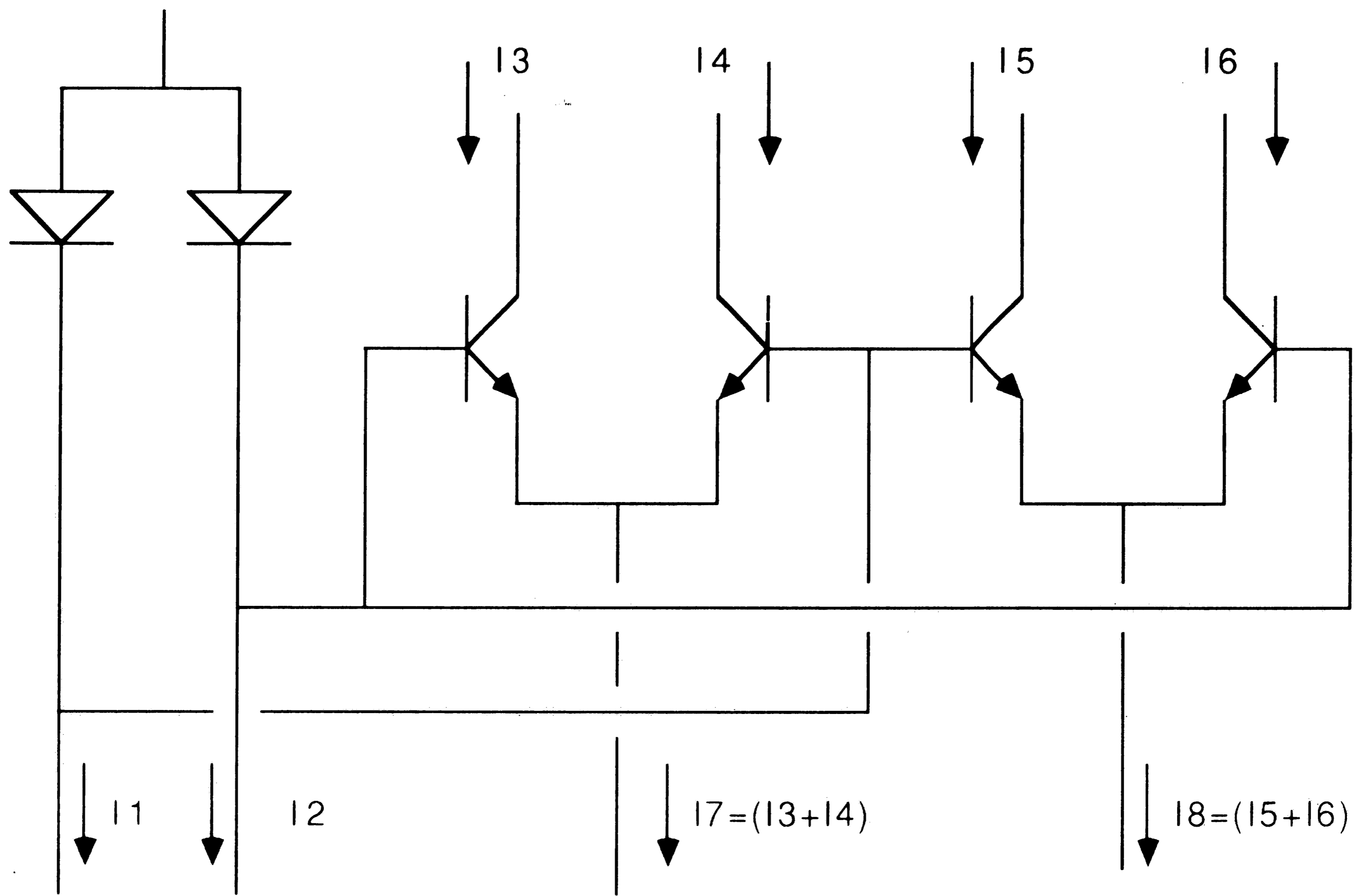


Figure 4.3

Square law configuration



## 4.2 SQUARE LAW CIRCUIT CONFIGURATION

Let now add another differential pair  $Q_5$   $Q_6$  to the circuit of figure 4.1.

From figure 4.3 , besides equation (4.2) we have:

$$I_6 - I_5 = \frac{(I_1 - I_2)(I_6 + I_5)}{(I_1 + I_2)} \quad (4.7)$$

From (4.2) and (4.7)

$$(I_3 - I_4) - (I_6 - I_5) = \frac{(I_1 - I_2)}{I_1 + I_2} (I_7 - I_8) \quad (4.8)$$

The difference  $I_7 - I_8$  can be obtained from input  $x$  in the same way of figure 4.2, thus equation (4.8) already indicates a square law characteristics.

A complete square law circuit is shown in figure 4.4

In figure 4.4,

$$I_{out} = \frac{(I_1 - I_2)(I_7 - I_8)}{(I_1 + I_2)} \quad (4.9)$$

$$I_{out} = 2 \frac{x^2}{R_1 R_2 I_x}$$

In figure 4.4,  $M_1$ ,  $M_2$ , and  $M_3$  are current mirrors, either Widlar type or for more accurate, Wilson type.

Note that we have something in common between  $Q_7$   $Q_8$  and  $Q_9$   $Q_{10}$  circuits, this permits us to simplify the circuit of figure 4.4 further as shown in figure 4.5.

The circuit basically remain the same, except for now the differential pairs  $Q_7$   $Q_8$  and  $Q_9$   $Q_{10}$  are driven by the same current



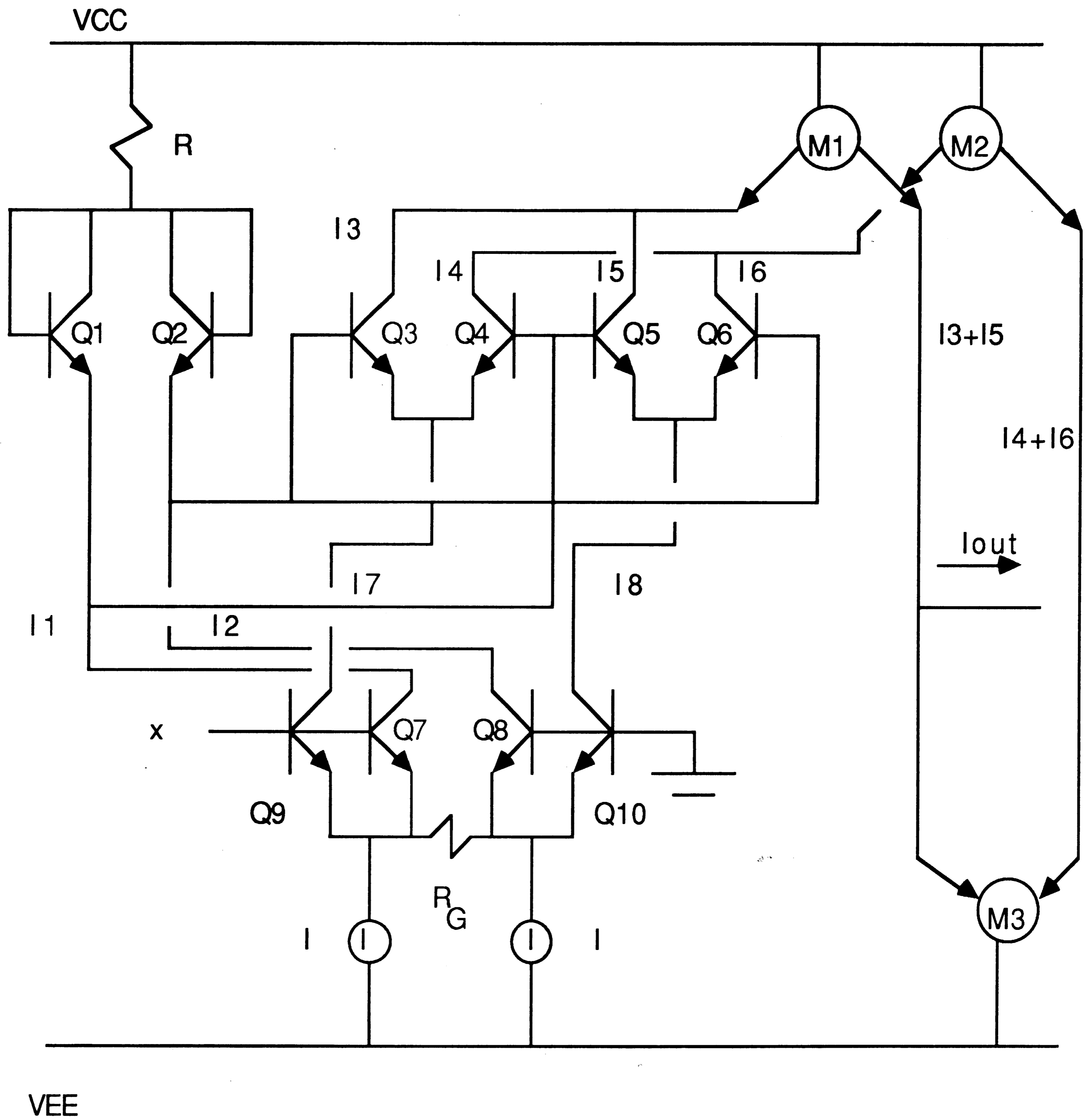


Figure 4.5

Simplified square law circuit

source I. In practice, transistors  $Q_7$  and  $Q_9$  can be merged to form a single transistor with double collectors. The same is applied for transistors  $Q_8$  and  $Q_{10}$

Again, ignore the base currents, we have

$$I_1 + I_7 = I + i$$

$$I_2 + I_8 = I - i$$

$$(I_1 + I_7) - (I_2 + I_8) = 2i$$

$$\text{Note also } I_1 = I_7 \text{ and } I_2 = I_8$$

Therefore

$$I_1 - I_2 = I_7 - I_8 = i = x/R_G$$

$$\text{And } I_1 + I_2 = I_7 + I_8 = I$$

The output current  $I_o$  of circuit in figure 4.5 now becomes

$$I_o = \frac{(I_1 - I_2)(I_7 - I_8)}{(I_1 + I_2)} \quad (4.10)$$

$$I_o = \frac{x^2}{R_G^2 I}$$

In general if we have  $n$  differential pairs connected as in figure 4.7, then it's easy to prove that :

$$I_1 - I_2 = I_3 - I_4 = \dots = I_{2n-1} - I_{2n} \quad (4.11)$$

$$I_{2n-1} - I_{2n} = \frac{2i}{n} = \frac{2x}{nR_G}$$

And

$$I_1 + I_2 = I_3 + I_4 = \dots = I_{2n-1} + I_{2n} \quad (4.12)$$

$$I_{2n-1} + I_{2n} = \frac{2I}{n}$$

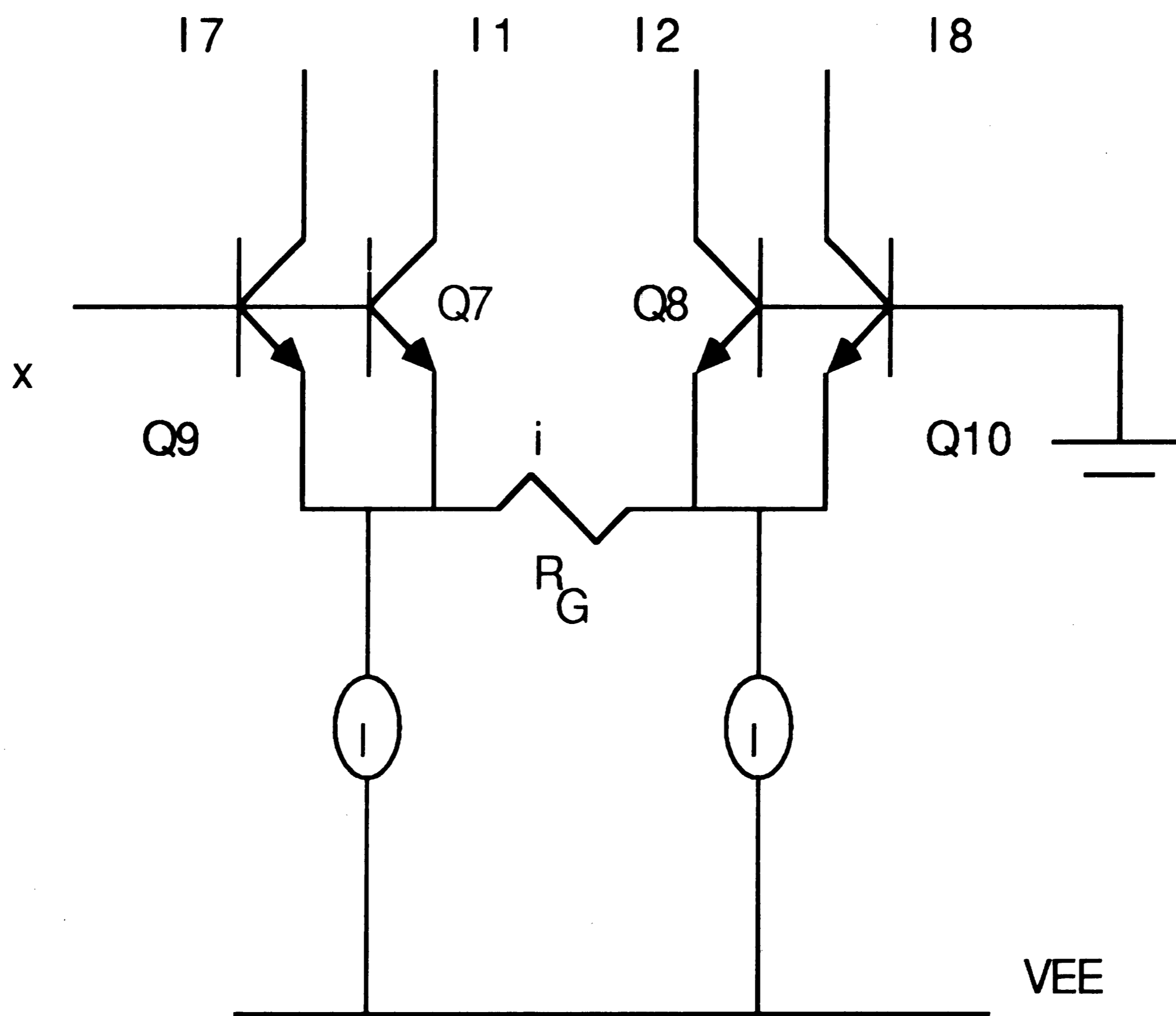


Figure 4.6  
Double differential V-I converter

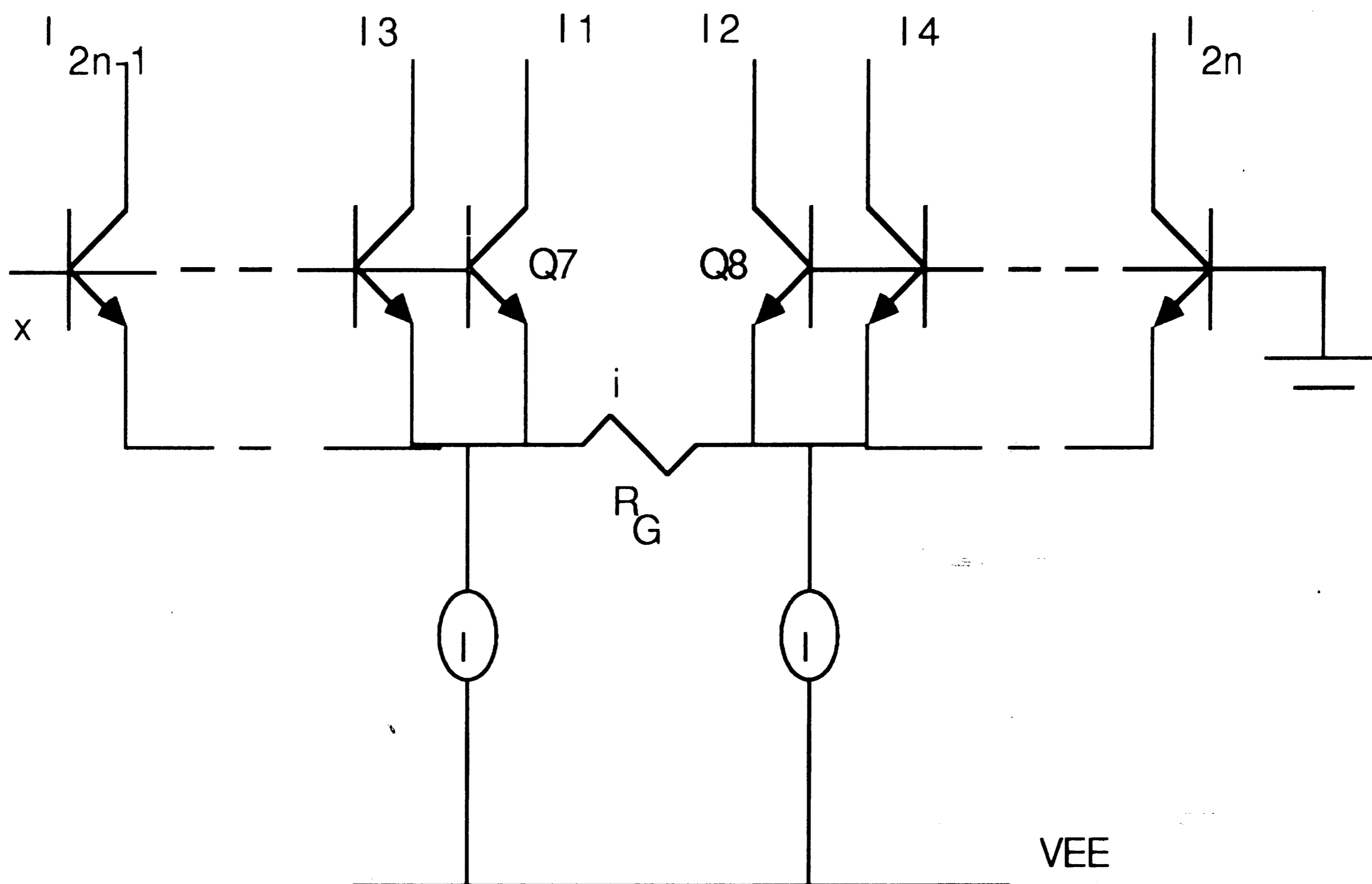


Figure 4.7 Multiple differential V-I converter

Equation (4.11) and (4.12) will be used later when we want to generate an  $n^{\text{th}}$  order polynomial .

### 4.3 POLYNOMIAL MODULE

An expansion of circuit in figure 4.5 will generate a cubic law function as well as any other higher order power function.

Let consider the circuit of figure 4.8

Apply the above results:

$$I_7 - I_8 = \frac{[I_1 - I_2][I_3 - I_4]}{[I_1 + I_2]}$$

$$I_9 - I_{10} = \frac{(I_7 - I_8)(I_5 - I_6)}{(I_7 + I_8)}$$

$$I_9 - I_{10} = \frac{(I_1 - I_2)(I_3 - I_4)(I_5 - I_6)}{(I_1 + I_2)(I_7 + I_8)}$$

$$I_9 - I_{10} = \frac{(I_1 - I_2)(I_3 - I_4)(I_5 - I_6)}{(I_1 + I_2)(I_3 + I_4)}$$

From equation (4.11) and (4.12) :

$$I_9 - I_{10} = \frac{[I_1 - I_2]^3}{[I_1 + I_2]^2}$$

(4.13)

$$I_9 - I_{10} = \frac{2x^3}{3(R_G)^3 I^2}$$

The differential output  $I_9 - I_{10}$  is easily converted into a single output by some additional current mirrors. Note that diode-connected  $Q_3$   $Q_4$  and  $Q_7$   $Q_8$  are ready to form current mirrors if we want to get 1st order and 2nd order power terms.

As it was mentioned above, an arbitrary order power term can be generated in the same manner. It's straight forward to find a

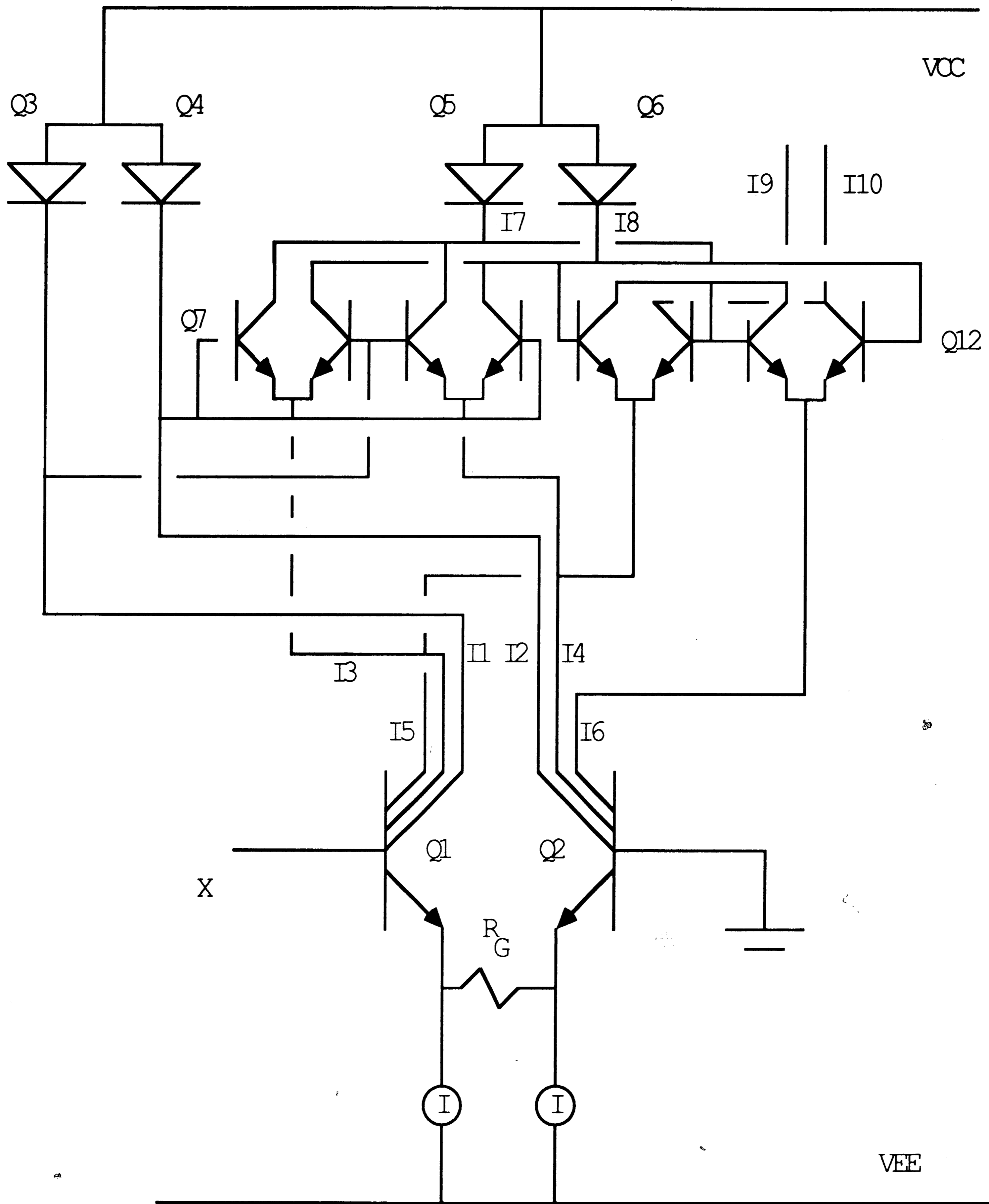


Figure 4.8

A cubic law circuit

closed form for a kth order power term.

$$I_{outk} = \frac{[I_1 - I_2]^k}{[I_1 + I_2]^{k-1}}$$

$$I_{outk} = \left(\frac{2x}{nR_G}\right)^k / \left(\frac{2I}{n}\right)^{k-1}$$

$$I_{outk} = \frac{2x^k}{n(R_G)^k I^{k-1}}$$

Denote

$$a_k = \frac{2}{n(R_G)^k I^{k-1}} \quad (4.14)$$

Then

$$I_{outk} = a_k x^k \quad (4.15)$$

In equation (4.14), n is the highest order available in the polynomial module. It will be set up depending on practical application of the module.

The polynomial will have following form

$$I_{out} = \sum_{k=1}^n I_{outk}$$

$$I_{out} = \sum_{k=1}^n a_k x^k$$

$$I_{out} = \sum_{k=1}^n \frac{2x^k}{n(R_G)^k I^{k-1}} \quad (4.16)$$



## 4.4 DESIGN CONSIDERATION

### 4.4.1 Voltage-current converter stage

The main cause of error is the logarithmic term that we omitted in equation (4.11)  $x = V_{BEQ_{2n-1}} + R_G i - V_{BEQ_{2n}}$

$$x = V_T \ln \frac{I_{2n-1}}{I_s} - V_T \ln \frac{I_{2n}}{I_s} + R_G i$$

$$x = V_T \ln \frac{I_{2n-1}}{I_{2n}} + R_G i$$

$$i = \frac{x}{R_G} - \frac{V_T}{R_G} \ln \frac{I_{2n-1}}{I_{2n}}$$

And

$$I_{2n-1} - I_{2n} = \frac{2i}{n}$$

$$I_{2n-1} - I_{2n} = \frac{2x}{nR_G} - \frac{2V_T}{nR_G} \ln \frac{I_{2n-1}}{I_{2n}}$$

The output current of a  $k^{\text{th}}$  order power has the form:

$$I_{outk} = \frac{[I_{2n-1} - I_{2n}]^k}{K}$$

Where  $K$  is a scale factor, depends upon resistor  $R_G$  and current source  $I$ .

$$I_{outk} = \left[ \frac{2x}{nR_G} - \frac{2V_T}{nR_G} \ln \frac{I_{2n-1}}{I_{2n}} \right]^k \frac{1}{K}$$

$$I_{ideal} = \left[ \frac{2x}{nR_G} \right]^k \frac{1}{K}$$

Denote error  $\varepsilon$ :

$$\varepsilon = \frac{I_{ideal} - I_{outk}}{I_{ideal}}$$

$$\varepsilon = \frac{(2x/nR_G)^k - [2x/nR_G - (2V_T/nR_G) \ln(I_{2n-1}/I_{2n})]^k}{(2x/nR_G)^k} \quad (4.17)$$

$$\varepsilon = 1 - \left[ 1 - \frac{V_T}{x} \ln \left[ \frac{I_{2n-1}}{I_{2n}} \right] \right]^k$$

We also have:

$$I_{2n-1} - I_{2n} \approx \frac{2x}{nR_G}$$

And:

$$I_{2n-1} + I_{2n} \approx \frac{2I}{n}$$

Therefore:

$$I_{2n-1} = \frac{I}{n} + \frac{x}{nR_G} \quad (4.18)$$

$$I_{2n} = \frac{I}{n} - \frac{x}{nR_G} \quad (4.19)$$

Replace (4.18) and (4.19) into (4.17):

$$\varepsilon = 1 - \left[ 1 - \frac{V_T}{x} \ln \left[ \frac{I/n + x/nR_G}{I/n - x/nR_G} \right] \right]^k$$

$$\varepsilon = 1 - \left[ 1 - \frac{V_T}{x} \ln \left[ \frac{InR_G + nx}{InR_G - nx} \right] \right]^k \quad (4.20)$$

$$\varepsilon = 1 - \left[ 1 - \frac{V_T}{x} \ln \left[ \frac{IR_G + x}{IR_G - x} \right] \right]^k$$

Figure 4.9a shows error  $\varepsilon$  as a function of input voltage  $x$  at different values of  $k$  and figure 4.9b shows  $\varepsilon$  as a function of  $k$  at a typical values of  $x$ .

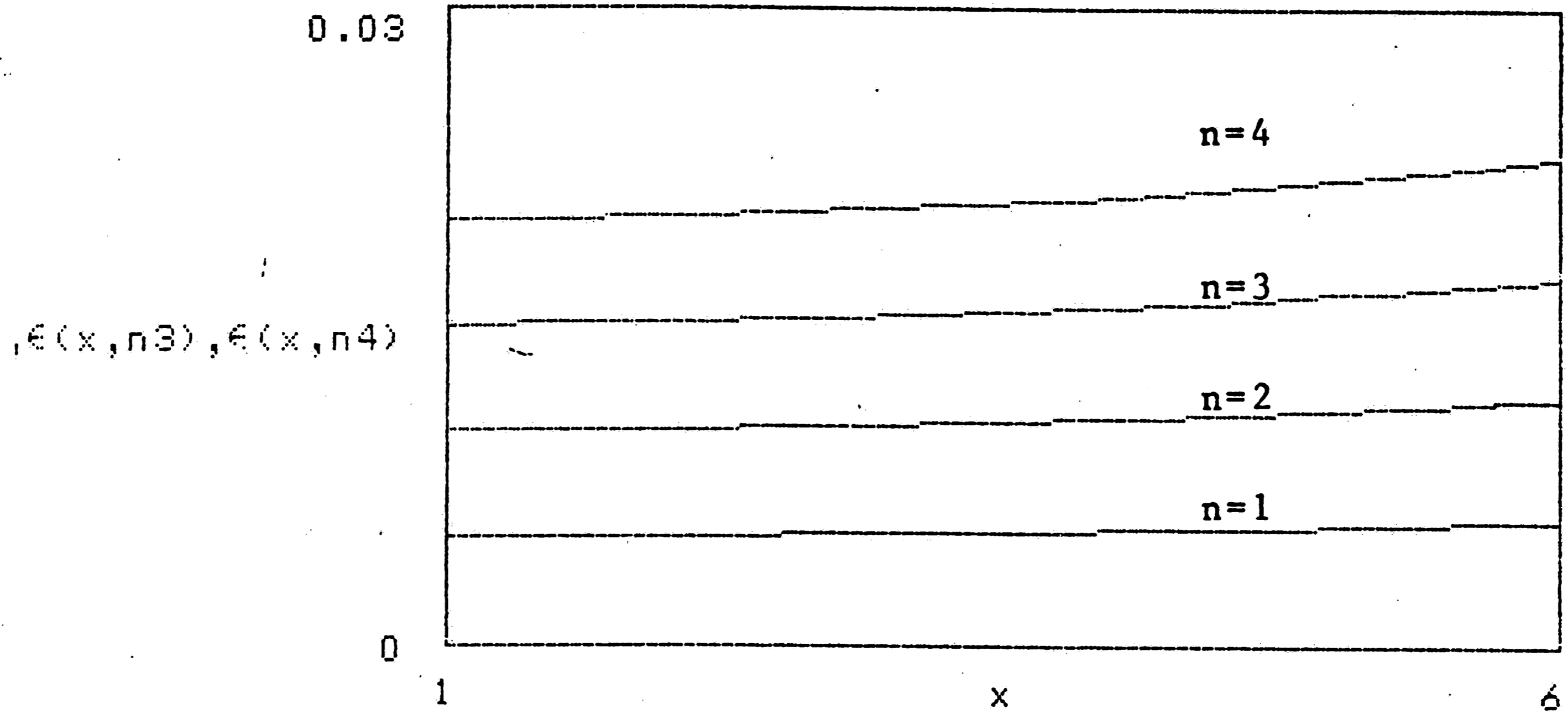


Figure 4.9a  
 $\epsilon$  as a function of input voltage  $x$  at different orders  $n$

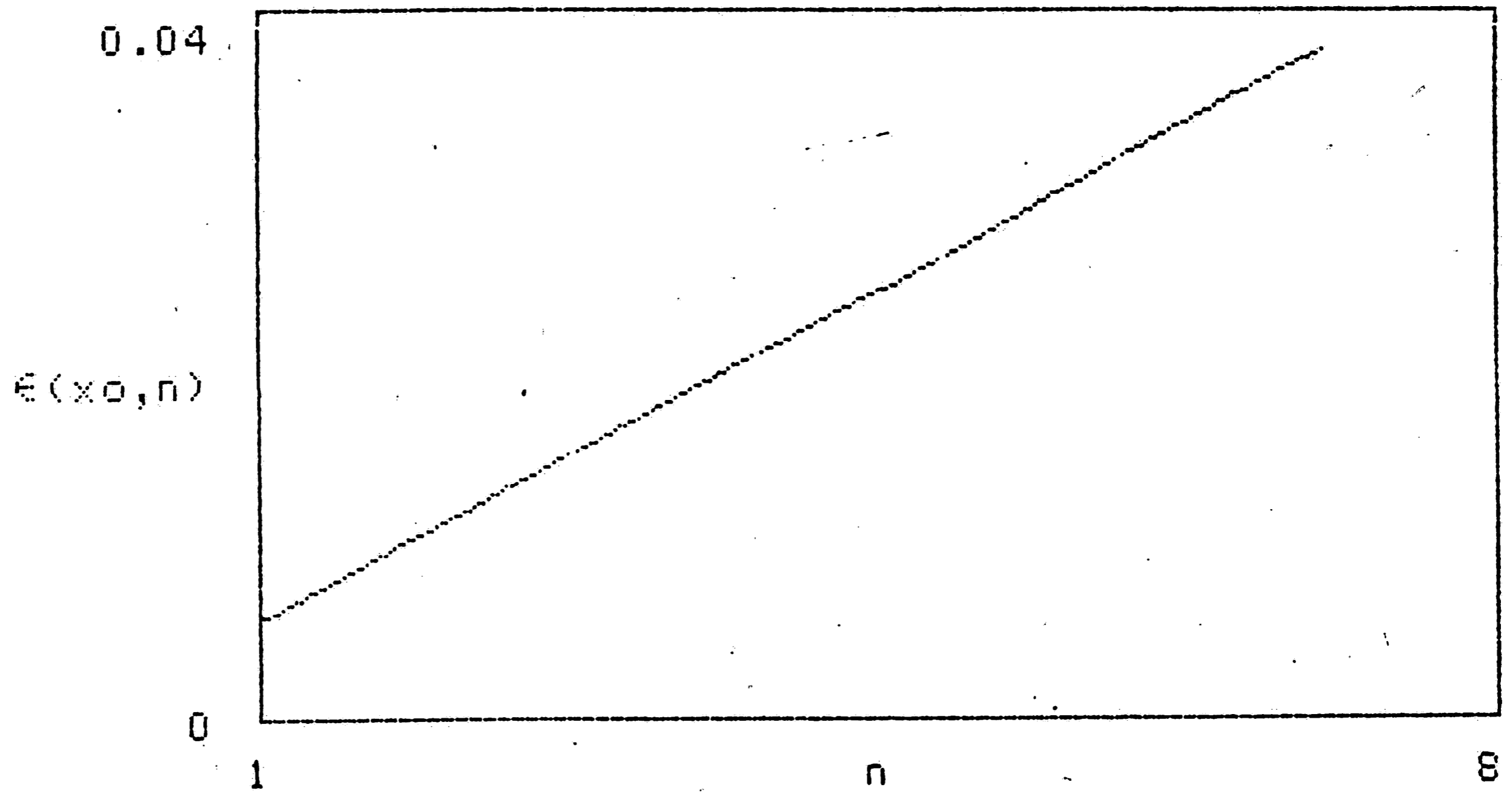


Figure 4.9b  
 $\epsilon$  as a function of  $n$  at a typical value of  $x$   
 ( $x=5$  volts)

$\epsilon$  is negligible when input voltage  $< 5$  Volts and  $n < 4$ .  $\epsilon$  increases for larger input voltage  $x$  and power order  $n$ . This error can be compensated either by trimming the gain resistor  $R_G$  or by using another transistor pair to cancel the affect of logarithmic term as in figure 4.10

In figure 4.10:

$$V_1 = V_{BEQ1} + V_{BEQ4} - R_G i - V_{BEQ3} - V_{BEQ2} + V_2$$

$$V_1 - V_2 = V_{BEQ1} - V_{BEQ3} + V_{BEQ4} - V_{BEQ2} - R_G i$$

$$V_1 - V_2 = V_T \ln \frac{I_1}{I_3} + V_T \ln \frac{I_2}{I_4} - R_G i$$

If  $Q_3$  and  $Q_4$  have fairly high  $\beta$ 's then  $I_1 = I_3$  and  $I_2 = I_4$ , therefore:

$$V_1 - V_2 = -R_G i$$

Thus the logarithmic term is canceled by  $Q_3$  and  $Q_4$ . In practice, to maintain transistors  $Q_3$  and  $Q_4$  in conduction region even with large change of differential input we can insert a diode Zener in the paths between the bases and collectors of  $Q_3$  and  $Q_4$ .

#### 4.4.2 Transistor mismatch

The second source of error is due to transistor mismatch in the multiplying cores. Let consider again the output of square term in the polynomial module in figure 4.11. Taking a closer look at the multiplying core  $Q_1$   $Q_2$   $Q_3$   $Q_4$  we have:

$$V_{BEQ1} + V_{BEQ4} = V_{BEQ2} + V_{BEQ3}$$

$$V_T \ln(I_1/I_{S1}) + V_T \ln(I_4/I_{S4}) = V_T \ln(I_2/I_{S2}) + V_T \ln(I_3/I_{S3})$$

$$\ln(I_1 I_{S2} / I_2 I_{S1}) = \ln(I_3 I_{S4} / I_4 I_{S3})$$

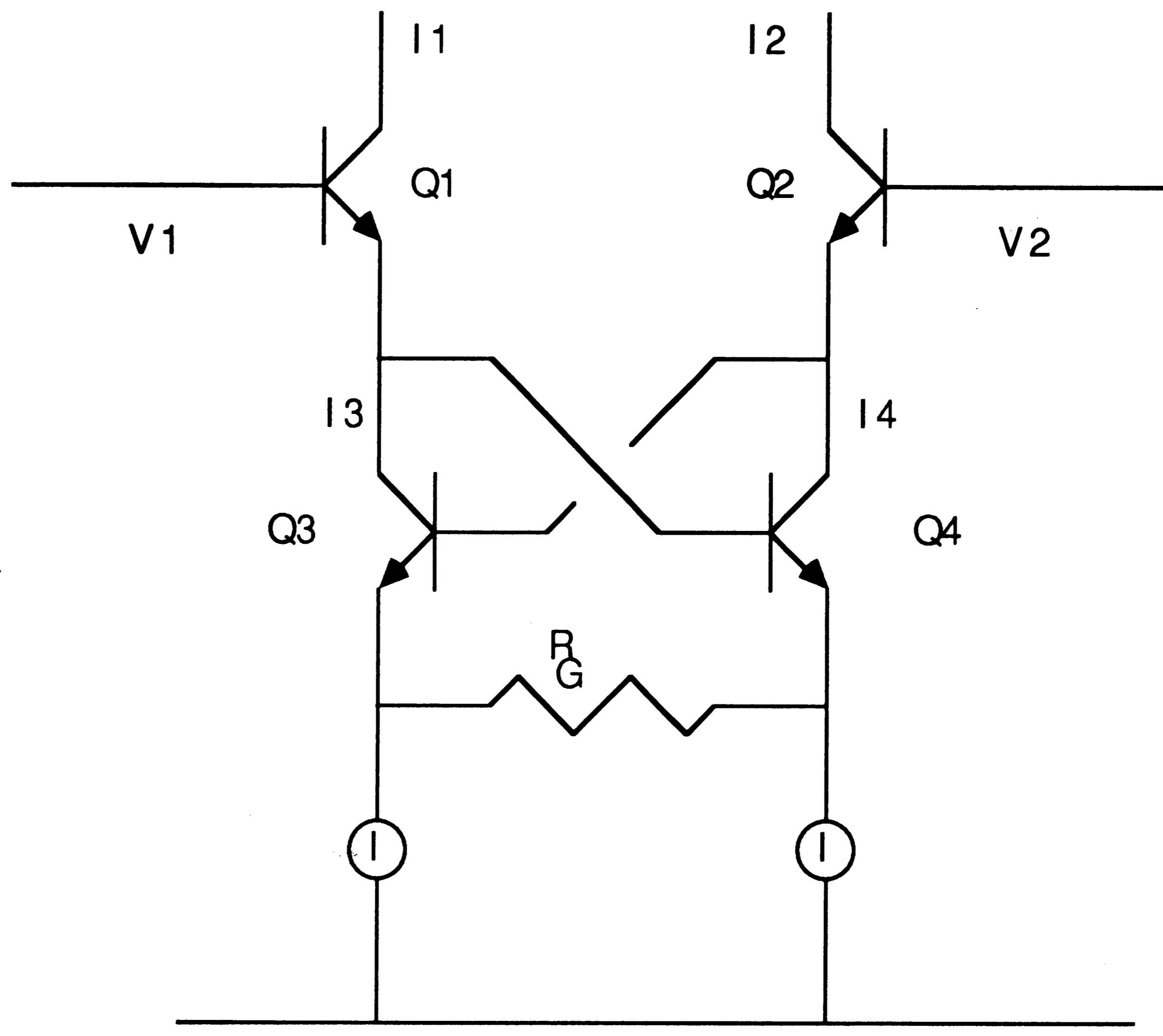


Figure 4.10

Compensated logarithmic error

Therefore

$$\frac{I_1 I_{S2}}{I_2 I_{S1}} = \frac{I_3 I_{S4}}{I_4 I_{S3}}$$

$$\frac{I_1}{I_2} = \frac{I_3}{I_4} \frac{I_{S1} I_{S4}}{I_{S2} I_{S3}}$$

Ideally if all transistors in the multiplying core are well matched, then the ratio of reverse bias saturation current  $I_S$ 's in the right-hand side of this equation would equal to 1. In a practical situation, we always have some degree of mismatch between transistors.

The reverse bias saturation current of a long-base p-n junction is defined by:

$$I_S = qA [D_p p_{no} / L_p + D_n n_{po} / L_n]$$

Where

q: electron charge

A: junction cross section area.

$D_p$ : diffusion coefficient of hole

$D_n$ : diffusion coefficient of electron

$p_{no}$ : minority carrier (hole) concentration at equilibrium in n region.

$n_{po}$ : minority carrier (electron) concentration at equilibrium in p region.

$L_p$ : hole diffusion length.

$L_n$ : electron diffusion length.

In these factors defining saturation current  $I_S$ , all of them except for junction cross section area can be well controlled

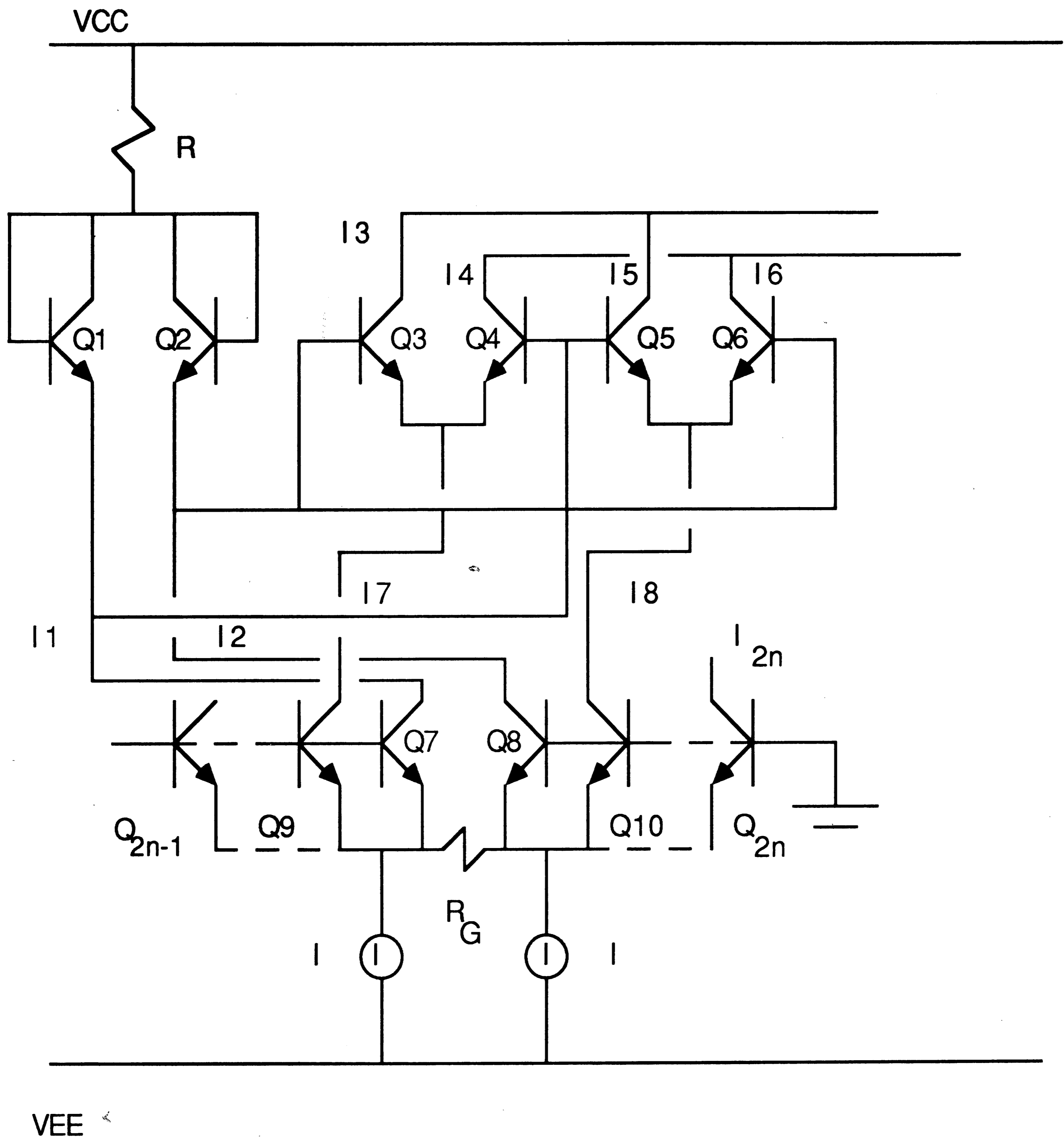


Figure 4.11

Square term output of the polynomial module

by fabrication process and their variation ,if any, will equally affect on every transistor. Thus the main source that cause  $I_S$  mismatch is the area mismatch of emitter-base junctions.

We can write

$$\frac{I_{S1}I_{S4}}{I_{S2}I_{S3}} = \frac{A_1A_4}{A_2A_3}$$

Where  $A_1$  to  $A_4$  are junction areas of  $Q_1$  to  $Q_4$

Denote the area mismatch factor

$$\alpha_1 = 1 - \frac{A_1A_4}{A_2A_3}$$

usually this is a very small number, within the range 2 percent. Now we have

$$\begin{aligned} \frac{I_1}{I_2} &= \frac{I_3(1-\alpha_1)}{I_4} \\ \frac{I_3(1-\alpha_1)}{I_1} &= \frac{I_4}{I_2} \\ I_3(1-\alpha_1) - I_4 &= \frac{I_1 - I_2}{I_1 + I_2} [I_3(1-\alpha_1) + I_4] \end{aligned} \quad (4.21)$$

The same arguement for the multiplying core  $Q_1$   $Q_2$   $Q_5$   $Q_6$  we have: (4.22)

$$I_6(1-\alpha_2) - I_5 = \frac{I_1 - I_2}{I_1 + I_2} [I_6(1-\alpha_2) + I_5]$$

Where

$$\alpha_2 = 1 - \frac{A_1A_5}{A_2A_6}$$

denotes the area mismatch of  $Q_1$   $Q_2$   $Q_5$   $Q_6$  core

From (4.21) and (4.22) we have



$$(I_3+I_5)-(I_4+I_6)+\alpha_2 I_6-\alpha_1 I_3$$

$$=\frac{I_1-I_2}{I_1+I_2}[(I_3+I_4)-(I_5+I_6)+\alpha_2 I_6-\alpha_1 I_3]$$

Or

$$(I_3+I_5)-(I_4+I_6)+\alpha_2 I_6-\alpha_1 I_3$$

$$=\frac{I_1-I_2}{I_1+I_2}(I_7-I_8)+\frac{I_1-I_2}{I_1+I_2}(\alpha_2 I_6-\alpha_1 I_3)$$

(4.23)

$$(I_3+I_5)-(I_4+I_6)=\frac{I_1-I_2}{I_1+I_2}(I_7-I_8)+(\alpha_2 I_6-\alpha_1 I_3)\left(\frac{I_1-I_2}{I_1+I_2}-1\right)$$

Note that we have an offset term

$$(\alpha_2 I_6-\alpha_1 I_3)\left(\frac{I_1-I_2}{I_1+I_2}-1\right)$$

compared with the ideal case. Obviously, in an ideal situation,  $\alpha_1=\alpha_2=0$  then equation (4.23) reduces to the usual form:

$$(I_3+I_5)-(I_4+I_6)=\left(\frac{I_1-I_2}{I_1+I_2}\right)(I_7-I_8)$$

Let  $\varepsilon$  denotes this offset term, we can express  $\varepsilon$  in terms of input voltage  $x$  and area mismatch factors  $\alpha_1$  and  $\alpha_2$

We have

$$\frac{I_3}{I_1} \approx \frac{I_4}{I_2} \approx \frac{I_3+I_4}{I_1+I_2} = \frac{I_7}{I_1+I_2}$$

From equation (4.11) and (4.12)

$$I_7 = \frac{I}{n} + \frac{x}{nR_G}$$

$$I_1+I_2 = \frac{2I}{n}$$

And hence

$$\frac{I_3}{I_1} = \frac{1}{2} + \frac{x}{2R_G I}$$

$$I_3 = \left(\frac{1}{2} + \frac{x}{2R_G I}\right) I_1$$

$$I_3 = \left(\frac{1}{2} + \frac{x}{2R_G I}\right) \left(\frac{I}{n} + \frac{x}{nR_G}\right)$$

Similarly:

$$I_6 = \left(\frac{1}{2} - \frac{x}{2R_G I}\right) \left(\frac{I}{n} + \frac{x}{nR_G}\right)$$

Also from (4.11) and (4.12):

$$\frac{I_1 - I_2}{I_1 + I_2} = \frac{x}{R_G I}$$

Replace this value and above values of  $I_3$  and  $I_6$  into the equation of  $\varepsilon$ , we have after simplifying:

$$\varepsilon = \frac{1}{n} \left[ \frac{I}{2} (\alpha_2 - \alpha_1) + \frac{(\alpha_1 + \alpha_2)}{2R_G} x + \frac{(\alpha_2 - \alpha_1)}{2(R_G)^2 I} x^2 - \frac{(\alpha_2 + \alpha_1)}{2(R_G)^3 I^2} x^3 \right] \quad (4.24)$$

Equation (4.24) shows that due to area mismatch we have a DC output offset of  $I(\alpha_2 - \alpha_1)/2$  and other distortions corresponding to higher order of input  $x$ . Equation (4.24) also shows that  $\varepsilon$  is reversely proportional to the highest order  $n$  of the polynomial module.

In practice, by careful layout, we can arrange the transistors in the multiplying cores such a way that  $\alpha_1$  and  $\alpha_2$  are very small and the error due to area mismatch can be limited within a negligible number.

Other errors caused by ohmic resistances at base-emitter junctions and by finite  $\beta$  are of second order effect, they are usually

very small . Transistors with large ohmic resistances and  $\beta$  in the neighborhood of 100 can be used without serious problem.

#### 4.4.3 Maximum input voltage swing

Our polynomial circuit works properly so long as all npn transistors sink current.

$$I_1 = I_3 = \dots = I_{2n-1} > 0$$

$$I_2 = I_4 = \dots = I_{2n} > 0$$

From equation (4.11) and (4.12)

$$I_1 = I_3 = \dots = I_{2n-1} = \frac{I}{n} + \frac{x}{nR_G}$$

$$I_2 = I_4 = \dots = I_{2n} = \frac{I}{n} - \frac{x}{nR_G}$$

The constraint for input voltage:

$$-IR_G < x < +IR_G$$

It is seen from equation (4.20) that when input  $x$  approaches the product  $IR_G$ , the logarithmic error caused by voltage-current converter stage increases significantly. As a rule of thumb, the maximum voltage swing  $x_{Max}$  is usually chosen such that

$$x_{Max} = g \cdot IR_G \tag{4.25}$$

Where  $g$  is dimensionless factor ranges from 0.5 to 0.8

#### 4.4.4 Maximum output current

The  $k^{th}$  term in the module:

$$I_{outk} = \frac{2x^k}{n(R_G)^k I^{k-1}}$$

$$I_{outk} = \frac{2x^k I}{n(R_G I)^k}$$

$$I_{outk} = g^k \frac{2x^k I}{n(x_{Max})^k}$$

Thus

$$I_{outkMax} = g^k \frac{2I(x_{Max})^k}{n(x_{Max})^k}$$

$$I_{outkMax} = g^k \frac{2I}{n} \tag{4.26}$$

$I_{outMax}$  depends on "tail" current  $I$ , the highest order  $n$  and the chosen factor  $g$ .

The choice for  $R_G$  and  $I$ , therefore, is decided by the range of the operating input voltage and the maximum current at the output according to equation (4.25) and equation (4.26).

#### 4.5 PROGRAMMING COEFFICIENTS $a_i$ 's

The module generates polynomial:

$$f(x) = \sum_{i=0}^n a_i x^i$$

As we have mentioned earlier, the module is used with the associated DSP network where any nonlinear device can be replaced by this module with certain allowable tolerance. The coefficients  $a_i$ 's may vary for each nonlinearity, therefore we need to provide some way to program coefficients  $a_i$ 's to make the module becomes more versatile.

Several schemes can be used to change a current gain, e.g, some variations of Widlar or Wilson current mirror; where the output current can be varied by mean of a potentionmeter. We prefer here to program the current gain, and hence the coefficients  $a_i$ 's, digitally by a binary code that will be entered by the user.

Let consider the R-2R network of figure 4.12a. To achieve current ratios at collectors, all transistors are kept at equal base-emitter voltages while emitter areas are binary ratioed.

$$\text{Ignore base currents : } V_{BEQ4} + 2RI_4 = V_{BEQ5} + 2RI_5 + R(I_5+I_6)$$

$$I_5 = I_6$$

$$V_T \ln \frac{I_4}{I_{S4}} + 2RI_4 = V_T \ln \frac{I_5}{I_{S5}} + 4RI_5$$

Or:

$$V_T \ln \frac{I_4 I_{S5}}{I_5 I_{S4}} = 2R(2I_5 - I_4)$$

$$I_5 = 2^{-1}I_4$$

$$\text{Similarly } I_4 = 2^{-1}I_3, \dots$$

We have:

$$I_2 = 2^{-1}I$$

$$I_3 = 2^{-2}I$$

$$I_4 = 2^{-3}I$$

$$I_5 = 2^{-4}I$$

This relationship holds to a high degree of accuracy thanks to the excellent matching and tracking characteristics of monolithic BJT's.

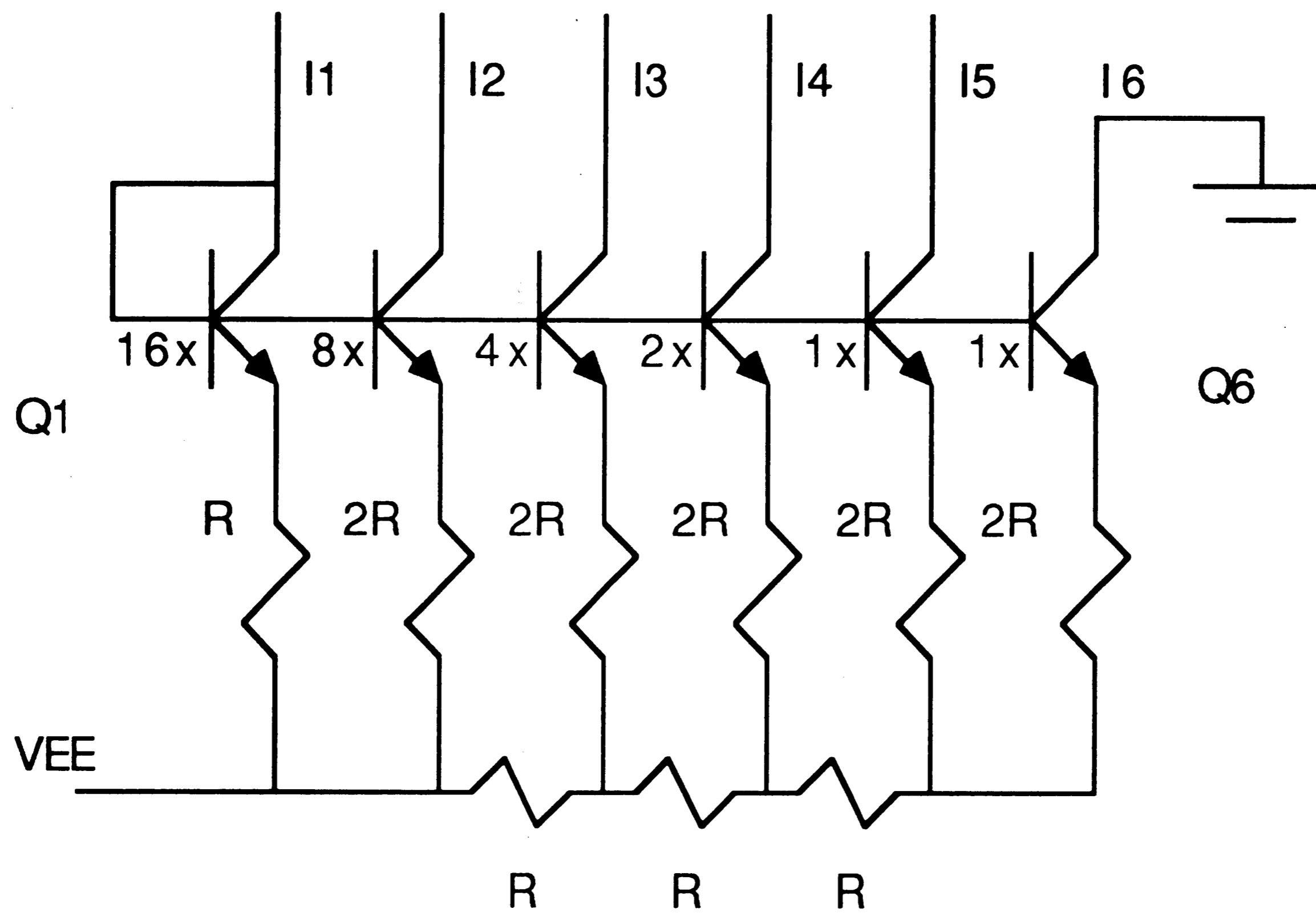


Figure 4.12a

R-2R current ladder

Now if we use a binary code to control switch  $S_1$  to  $S_4$  as in figure 4.12b, we can program the output current  $I_o$ .

$$I_o = b_1 I_2 + b_2 I_3 + b_3 I_4 + b_4 I_5$$

In general:

$$I_o = \sum_{i=1}^n b_i 2^{-i} \quad (4.27)$$

The circuit of figure 4.12b can only sink current  $I_o$ , to program  $I_o$  in both directions, we may use the circuit of figure 4.13 in which another bit,  $b_{sign}$ , was introduced to control the sign ( or direction ) of output current  $I_o$ .

$$I_{sign} = \begin{cases} I_{in} & \text{if } b_{sign} = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$I_{out} = I_o - I_{sign}$$

(4.28)

$$I_{out} = I_{in} \left( \sum_{i=1}^n b_i 2^{-i} - b_{sign} \right)$$

Equation (4.28) shows that by an additional sign bit , we can program the output current in both directions. This Programmable Current Amplifier (PCA) can be used to program coefficients  $a_i$ 's of the polynomial module.

One drawback of this circuit is the input current can only flow into the programming circuit; this may cause problem. As an alternative , we can use the current mode multiplying DAC type of

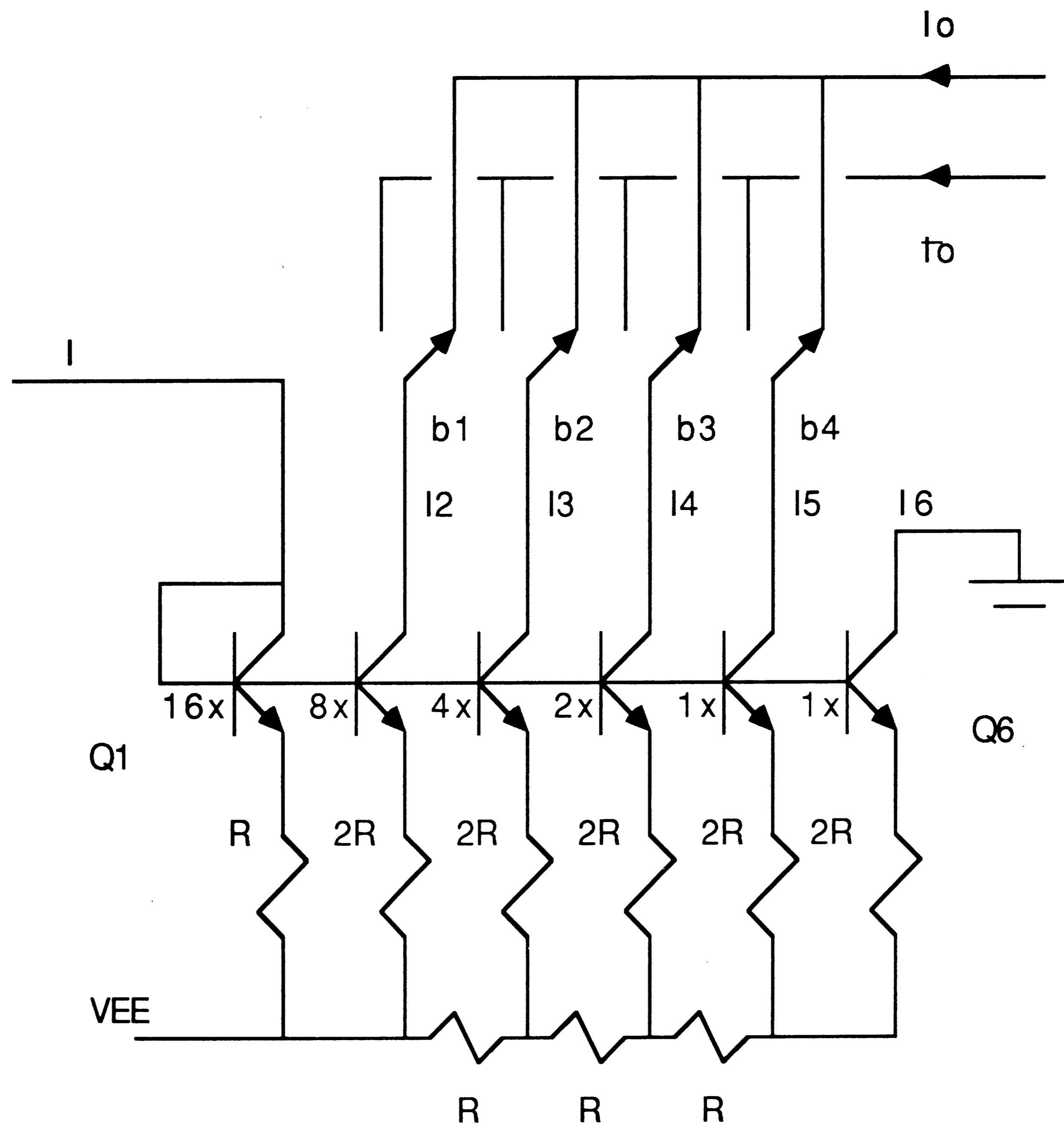


Figure 4.12b  
 Multiplying 4 bit R-2R DAC



figure 3.5 in chapter3, where the input current can flow in both directions.

Figure 4.14 shows a polynomial module with programmable coefficients. M's are current mirrors, to avoid an output offset current Wilson current mirrors should be used.

PCA1 , PCA2, PCA3 are Programmable Current Amplifiers. Each PCA can be programmed independently by different programming codes. Each programming code includes a sign bit  $b_s$  and other multiplying bit  $b_i$ 's.



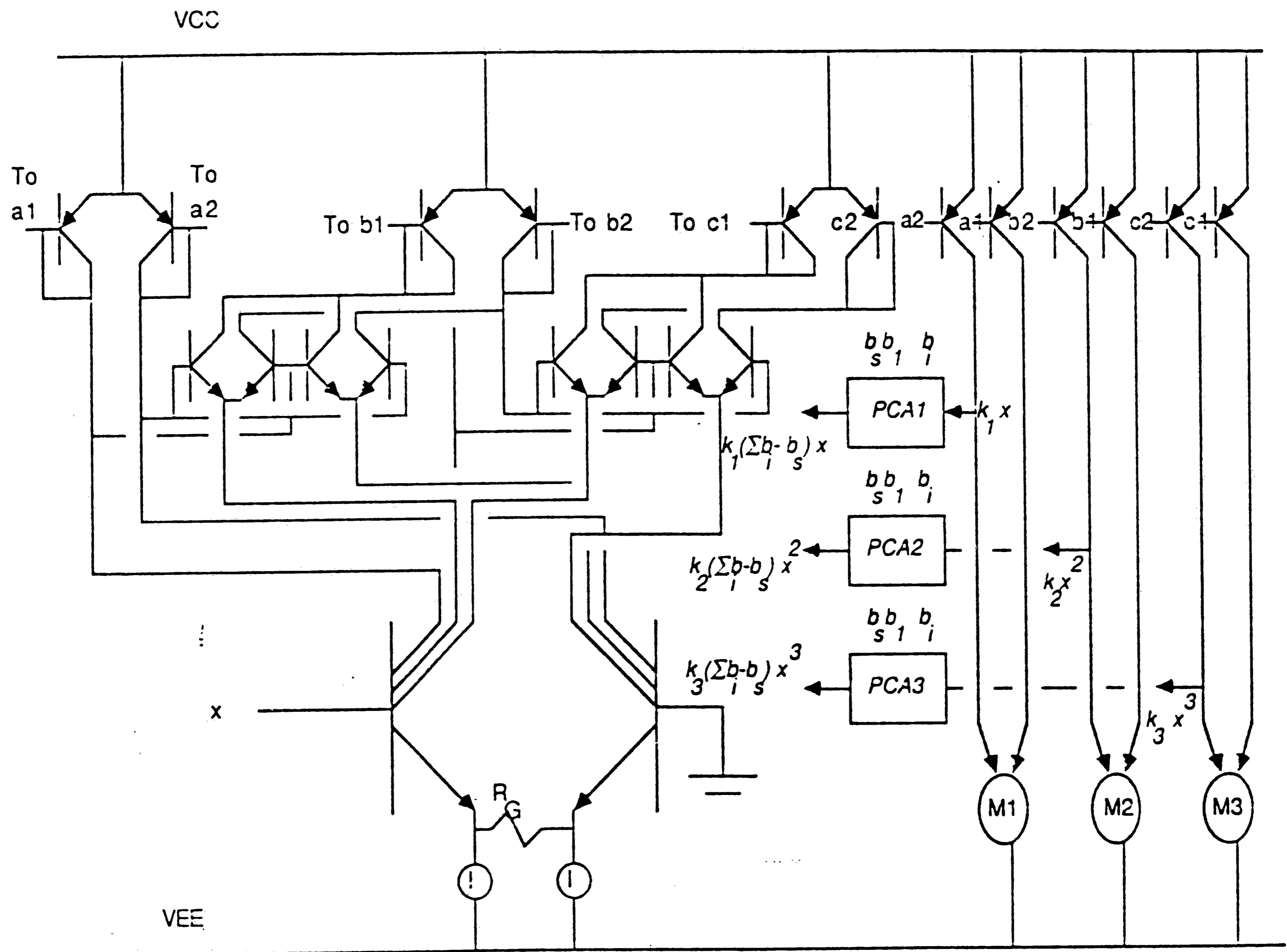


Figure 4.14

Programmable Polynomial Module

# **Chapter 5**

## **Existence and uniqueness of solution for the equation $\mathbf{x}=\mathbf{w}-\mathbf{B}\mathbf{F}(\mathbf{x})$ of nonlinear networks**

In chapter 3 we have explored several circuit topologies to realize equation  $\mathbf{x}=\mathbf{w}-\mathbf{B}\mathbf{F}(\mathbf{x})$  wherein we inherently have assumed that this standard equation converges to a solution. This is not always the case. Depending on nonlinear mapping  $\mathbf{F}(\mathbf{x})$  and the network topology  $\mathbf{B}$ , the standard equation may give no solution or it may give more than one solution.

In general, the study for existence and uniqueness of solution for the equation of nonlinear networks involves many issues and in fact, so far there is no generally applicable theory for the determination of nonlinear network solution. Dealing with these issues in detail, therefore, is beyond the scope of this work. What we hope here is that by employing several recent contributions to the problem of nonlinear network solution we may derive some results applied for our network equation.

### **5.1 BACKGROUND**

Several attempts have been devoted to provide a reasonably comprehensive study of solution for nonlinear networks. As early as in the years of forties, nonlinear resistive networks were studied by Duffin [12]. His basic theorem stated that a network of nonlinear resistors, each of which is characterized by a continuous strictly

monotone increasing function that map the real line onto itself, and independent voltage and current sources has a unique solution.

Duffin's treatment of existence was limited to certain case of strictly monotone nonlinear resistive networks and the result of his work is far from being a necessary condition, because it's clear that many resistive networks containing voltage or current- controlled elements whose characteristics saturate also have a unique solution. This occurs when the network's topology is such that the noninvertibility of the i-v characteristic functions of certain elements is of no consequence.

Since saturating i-v characteristics are often used in models of common two-terminal semiconductor devices, it's important to be able to identify these networks. Some early attempts at dealing with such networks are contained in papers by Desoer and Katzenelson [13], and later by Desoer and Wu [14]. Their studies concerned in physical structure of the nonlinear networks, and included some topological conditions for the existence and uniqueness of network solution.

Recently, several results have been contributed to this issue by Wilson and Sandberg in their papers [9], [15]. Concerning nonlinear network analysis they have shown that for an n-port resistive network, the problem of determining a solution for the network is equivalent to the problem of solving equation  $AF(x)+Bx=c$  where  $k=1,2,\dots,n$  the component  $x_k$  of the vector  $x=(x_1,x_2,\dots,x_n)^T$  corresponds to the port variable at the  $k^{\text{th}}$  port.  $x_k$  is the controlling

variable of the  $k^{\text{th}}$  nonlinear resistor. The nonlinear mapping  $F$  characterizes the nonlinear resistors and is defined, for all  $n$ -vector  $x$ , by  $F(x) = (f_1(x_1), f_2(x_2), \dots, f_n(x_n))^T$ .  $A$  and  $B$  are  $n \times n$  matrices of real number and  $c$  denotes a real  $n$ -vector.  $A$  and  $B$  provide the characterization of the linear portion of the network. Wilson and Sandberg have made significant advance to the analysis of nonlinear resistive networks. Two following theorems of their work are reproduced here because it is somewhat related to the study of our network equation

### **Sandberg and Wilson's 1st theorem**

Let  $F \in \mu^n$  be the set of all strictly increasing functions mapping  $R^n$  into itself, and let  $(A, B)$  be a passive pair of real  $n \times n$  matrices, then there exists a unique solution of  $AF(x) + Bx = C$  (1) for each  $C$  in  $R^n$  if and only if  $\beta(F) \cap N(B) = \{\theta\}$ , if

$$\beta(F) \cap N(B) \neq \{\theta\}$$

then there exists some  $C \in R^n$  such that (1) has no solution.

In there theorem,  $\beta(F)$  is the set of all points  $x$  in  $R^n$  for which  $F[x]$  is bounded as  $x \rightarrow \infty$

$N(B)$  is the null space of  $B$ , the set of all real  $n$ -vector  $x$  such  $Bx = \theta$ , where  $\theta$  denotes the origin of  $n$ -space  $R^n$

A pair  $(A, B)$  is called a passive pair if it possesses the following property: for each pair of  $n$ -vector  $(x, y)$  satisfying  $Ax = By$  it follows that

$$x^T \cdot y \geq 0.$$

According to Sandberg and Wilson when all linear resistors in the  $n$ -

port network have non-negative resistance or conductance values then the matrices A and B associated with the n-port network contains a passive pair (A,B).

The notation

$$\beta(F) \cap N(B) = \{\theta\}$$

simply means that there exists no real n-vector  $x \neq \theta$  for which  $Bx = \theta$  and  $\lim_{\rho \rightarrow \infty} |F[\rho x]| < \infty$ .

The nature of the matrix B sets specific limitations on the manner in which the nonlinear resistor i-v characteristic functions  $f_k$  are permitted to saturate. The relation

$$\beta(F) \cap N(B)$$

relates the nature of nonlinear resistor i-v characteristic functions  $f_k$  to the pertinent aspects of the topological structure of the network represented by matrix B. From the result of Desoer and Wu, the relation  $\beta(F) \cap N(B) = \{\theta\}$  will be satisfied once conditions for topological structure in Desoer and Wu's theorem are satisfied.

For networks containing resistors whose characteristics are not necessarily monotone-increasing, Sandberg and Wilson have shown that:

### **Sandberg and Wilson's 2nd theorem**

Let F be a nonlinear mapping with all the components  $f_k$  are eventually strictly increasing, let (A,B) be a passive pair of real nxn matrices. Then there exists at least one solution of (1) for each real n-vector C if  $\beta(F) \cap N(B) = \{\theta\}$

In this theorem, eventually strictly increasing function is a

function that is strictly increasing at large values of controlling variable.

## **5.2 EXISTENCE AND UNIQUENESS OF SOLUTION FOR EQUATION $\mathbf{x}=\mathbf{w}-\mathbf{B}\mathbf{F}(\mathbf{x})$**

We now use the above results to develop some criteria for the existence and uniqueness of solution of our equation

$$\mathbf{x}=\mathbf{w}-\mathbf{B}\mathbf{F}(\mathbf{x}) \quad (5.1)$$

Recall that  $\mathbf{x}$  and  $\mathbf{w}$  are real  $n$ -vectors,  $\mathbf{F}(\mathbf{x})$  is a nonlinear mapping which maps  $\mathbb{R}^n$  into  $\mathbb{R}^n$  such that  $\mathbf{F}(\mathbf{x})=[f_1(x_1), f_2(x_2), \dots, f_n(x_n)]^T$ , and  $\mathbf{B}$  is real  $n \times n$  matrix determined by the network topology.

Assume that matrix  $\mathbf{B}$  is non-singular, i.e.  $\det(\mathbf{B}) \neq 0$ , then equation (5.1) is equivalent to:

$$\mathbf{A}\mathbf{x}+\mathbf{F}(\mathbf{x})=\mathbf{C} \quad (5.2)$$

Where

$$\mathbf{A}=\mathbf{B}^{-1}$$

$$\mathbf{C}=\mathbf{B}^{-1}.\mathbf{w}$$

### **5.2.1 Nonlinearities with strictly monotone-increasing characteristics.**

#### **5.2.1.1 Sufficient condition for the uniqueness of solution for**

$$\mathbf{x}=\mathbf{w}-\mathbf{B}\mathbf{F}(\mathbf{x})$$

When all the nonlinearities are characterized by strictly increasing functions, we can apply the following theorem proven by Wilson in [16]

#### **Wilson's theorem**



Equation  $Ax+F(x)=C$  possesses a unique solution whenever two following conditions are satisfied:

(i) All components of  $F(x)$  are strictly monotone-increasing function, mapping from  $R$  onto  $R$

(ii) Matrix  $A$  is a weakly row-sum dominant matrix, i.e:

$$a_{ii} \geq \sum_{j=1, j \neq i}^n |a_{ij}|$$

for  $i=1,2,\dots,n$

By relating the theorem of Wilson for equation (5.2) we can immediately derive the following result for equation (5.1):

**Result 1**

*Equation  $x=w-BF(x)$  possesses a unique solution for every  $w$  in  $R^n$  whenever two following conditions are satisfied:*

(i). *All the components  $f_j(x)$ ,  $j=1,2,\dots,n$  of  $F(x)$  are strictly increasing functions, mapping the real line onto itself.*

(ii). *Determinant of matrix  $B$ ,  $\det(B) \neq 0$  and  $B^{-1}$  is a weakly row-sum dominant matrix.*

In result 1, condition  $\det(B) \neq 0$  is necessary to guarantee the existence of  $B^{-1}$ . The above result is stated as a sufficient condition. A condition which is both sufficient and necessary will be stated in term of a class of Po matrices.

**5.2.1.2 Necessary and sufficient condition for the uniqueness  
of solution for equation  $x=w-BF(x)$**

**Class Po matrix definition**

The class of matrix A satisfying one of the following equivalent conditions is denoted by Po :

(i).All principal minors of A are non-negative

(ii).For each vector  $x \neq \theta$ , there exists a index k such that

$$x_k \neq 0$$

and  $x_k y_k > 0$  where  $y=Ax$

(iii).For each vector  $x \neq \theta$  there exists a diagonal matrix  $D_x \geq \theta$  such that  $\langle x, D_x \cdot x \rangle$  and  $\langle Ax, D_x \cdot x \rangle$  are greater than or equal 0.

Where  $\langle x, y \rangle$  denotes the inner product of vector x and y,

$$\langle x, y \rangle = \sum_{i=1}^n x_i y_i$$

(iv).Every real Eigenvalue of A as well as of each principal submatrix of A is non-negative.

(v).For every diagonal matrix  $D > \theta$

$$\det(D+A) \neq 0$$

In the above properties , a principal submatrix of square matrix A is any square submatrix of A whose main diagonal is contained in the main diagonal of A.

The determinant of a principal submatrix is called principal minor.

**Class of function definition**

1-For all  $\alpha, \beta$  with

$$-\infty \leq \alpha < \beta \leq +\infty$$

let  $I(\alpha, \beta)$  denotes the interval :

$$I(\alpha, \beta) = \{x : \alpha < x < \beta\}$$

2-For each positive integer  $n$  and each pair of  $n$ -vectors  $\alpha, \beta$  let  $f^n(\alpha, \beta; R^n)$  denotes the set of mappings from  $I(\alpha_1, \beta_1) \times I(\alpha_2, \beta_2) \times \dots \times I(\alpha_n, \beta_n)$  onto  $R^n$  defined by  $F$  is in  $f^n(\alpha, \beta; R^n)$  if and only if there exists, for  $i=1, 2, \dots, n$ , strictly increasing functions  $f_i$  mapping  $(\alpha_i, \beta_i)$  onto  $R^1$  such that for  $x=(x_1, x_2, \dots, x_n)^T$ ,  $F(x)=[f_1(x_1), f_2(x_2), \dots, f_n(x_n)]^T$ .

3-Let the set of strictly increasing mappings from  $R^n$  onto  $I(\alpha_1, \beta_1) \times I(\alpha_2, \beta_2) \times \dots \times I(\alpha_n, \beta_n)$  be similarly defined and denoted  $f^n(R^n; \alpha, \beta)$ .

In the above definition,  $I(\alpha_1, \beta_1) \times I(\alpha_2, \beta_2) \times \dots \times I(\alpha_n, \beta_n)$  denotes the Cartesian product of closed interval in  $R^1$ . Also note that  $F$  is in  $f^n(\alpha, \beta; R^n)$  if and only if  $F^{-1}$  is in  $f^n(R^n; \alpha, \beta)$

Sandberg and Wilson proved in [17] that:

### **Sandberg and Wilson 's theorem**

There exists a unique solution of  $F(x)+Ax=C$  for each  $F$  in  $f^n(\alpha, \beta; R^n)$  and each  $C$  in  $R^n$  if and only if  $A$  is in  $P_0$ .

Apply for our equation, we have a similar result:

### **Result 2**

*There exists a unique solution for equation  $x=w-BF(x)$  for every  $F$  in  $f^n(\alpha, \beta; R^n)$  and every  $w$  in  $R^n$  if and only if  $B$  is in  $P_0$  and  $\det(B) \neq 0$ .*

Again, the additional condition  $\det(B) \neq 0$  to guarantee  $B^{-1}$  exists. Also notice that, according to Sandberg and Wilson, if  $B \in P_0$

and  $\det(B) \neq 0$  then  $B^{-1} \in P_0$

When the mapping  $f(\cdot)$  is in  $f^n(R^n; \alpha, \beta)$  then we can easily prove the following statement:

**Result 3**

*There exists a unique solution for equation  $x = w - BF(x)$  for every  $F$  in  $f^n(R^n; \alpha, \beta)$  and every  $w$  in  $R^n$  if and only if  $B$  is in  $P_0$ .*

Proof of result 3:

As it was mentioned earlier if  $F \in f^n(R^n; \alpha, \beta)$  then  $F^{-1}$  exists and  $F^{-1} \in f^n(\alpha, \beta; R^n)$

From equation  $x + BF(x) = w$  (1)

We have the equivalent equation:  $F^{-1}(y) + By = w$  (2)

According to the above theorem of Sandberg and Wilson, equation (2) has a unique solution if and only if  $B \in P_0$ , so does equation (1)

### 5.2.2 Nonlinearities with non-monotone characteristics

In the previous section we have studied the standard equation  $x + BF(x) = w$  in which the functions  $f_j$ 's are strictly monotone increasing. For such a situation the conditions for the solution's uniqueness had already been established. We now consider equations in which the nonlinear functions  $f_j$ 's are continuous but not necessarily monotone. In contrast to the monotone case, we have known at the beginning of this chapter that in such situations the equation may possess more than one solution. We will restrict ourselves to study here only nonlinearities with eventually strictly

increasing characteristics. An eventually strictly increasing function is a function that is strictly increasing for all large values of variable, i-v characteristics of a tunnel-diode is a typical example of this type of function.

In [4] Sandberg and Wilson have shown that the equation  $AF(x) + Bx = C$ , in which  $F(x)$  composed of eventually strictly increasing functions has at least one solution if :

(i). For each diagonal matrix  $D = \text{diag}(d_1, d_2, \dots, d_n)$  with each  $k$ ,  $d_k = +1$  or  $-1$ , there exists a real n-vector  $p$  such that

$$DA^T p \geq \theta$$

$$DB^T p \geq \theta$$

$$D(A+B)^T p \geq \theta$$

Where  $\theta$  is the origin of  $R^n$

(ii). And

$$\beta(F) \cap N(B) = \{\theta\}$$

We can apply the above result of Sandberg and Wilson to derive a condition for our equation  $x + BF(x) = w$

$$\text{Rewrite } BF(x) + Ix = w$$

Where  $I$  is the identity matrix.

Obviously the null space of  $I$ ,  $N(I) = \{\theta\}$ , therefore condition (ii) of Sandberg and Wilson's theorem is immediately satisfied. Now we consider condition (i).

Let  $D$  be a diagonal matrix with entries are either  $+1$ 's or  $-1$ 's.

Let  $p$  be a real n-vector.

$$\text{Denote } M = DB^T p$$

$$M = \begin{bmatrix} d_1 & 0 & \dots & 0 \\ 0 & d_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & d_n \end{bmatrix} \begin{bmatrix} b_{11} & b_{21} & \dots & b_{n1} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \dots & b_{nn} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}$$

$$= \begin{bmatrix} d_1 p_1 b_{11} + d_1 p_2 b_{21} + \dots + d_1 p_n b_{n1} \\ d_2 p_1 b_{12} + d_2 p_2 b_{22} + \dots + d_2 p_n b_{n2} \\ \vdots \\ d_n p_1 b_{1n} + d_n p_2 b_{2n} + \dots + d_n p_n b_{nn} \end{bmatrix}$$

If we choose  $p_1 = d_1$

$$p_2 = d_2$$

$$p_n = d_n$$

Then

$$M = \begin{bmatrix} d_1^2 b_{11} + d_1 d_2 b_{21} + \dots + d_1 d_n b_{n1} \\ d_2 d_1 b_{12} + d_2^2 b_{22} + \dots + d_2 d_n b_{n2} \\ \vdots \\ d_n d_1 b_{1n} + d_n d_2 b_{2n} + \dots + d_n^2 b_{nn} \end{bmatrix} \quad (1)$$

Because  $d_k$  is equal either +1 or -1, therefore:

$$M > = \begin{bmatrix} b_{11} - \{ |b_{21}| + \dots + |b_{n1}| \} \\ b_{22} - \{ |b_{12}| + |b_{32}| + \dots + |b_{n2}| \} \\ \vdots \\ b_{nn} - \{ |b_{1n}| + |b_{2n}| + \dots + |b_{n-1,n}| \} \end{bmatrix}$$

Thus if the matrix B has the property that :

$$b_{jj} \geq \sum_{i=1, i \neq j}^n |b_{ij}|$$

That is B is a weakly column-sum dominant matrix, then  $M \geq \theta$

2- denote  $N = DI^T p = Dp$

$$N = \begin{vmatrix} d_1 & 0 & \dots & 0 \\ 0 & d_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & d_n \end{vmatrix} \begin{vmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{vmatrix} = \begin{vmatrix} d_1 p_1 \\ d_2 p_2 \\ \vdots \\ d_n p_n \end{vmatrix}$$

The vector that we have chosen,  $p_i = d_i$ , therefore  $N \geq \theta$

3-Denote  $Q = D(B+I)^T p$

$$Q = \begin{vmatrix} d_1 & 0 & \dots & 0 \\ 0 & d_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & d_n \end{vmatrix} \begin{vmatrix} b_{11}+1 & b_{21} & \dots & b_{n1} \\ b_{12} & b_{22}+1 & \dots & b_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ b_{1n} & b_{2n} & \dots & b_{nn}+1 \end{vmatrix} \begin{vmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{vmatrix}$$

$$Q = \begin{vmatrix} d_1^2(b_{11}+1) + d_1 d_2 b_{21} + \dots + d_1 d_n b_{n1} \\ d_2 d_1 b_{12} + d_2^2(b_{22}+1) + \dots + d_2 d_n b_{n2} \\ \vdots \\ d_n d_1 b_{1n} + d_n d_2 b_{2n} + \dots + d_n^2(b_{nn}+1) \end{vmatrix} \quad (\text{II})$$

Compare (I) and (II) we have:  $Q > M$

Therefore:  $Q > \theta$  if  $b$  is weakly column-sum dominant.

We arrive to following result which can be considered as a corollary of Sandberg and Wilson's theorem:

#### Result 4

*Equation  $x = w - BF(x)$  in which  $F(x)$  consists of eventually strictly monotone increasing functions mapping from  $R^n$  into  $R^n$  possesses at least one solution for every  $w \in R^n$  if  $B$  is a weakly column-sum dominant matrix.*

From the above results, we have the following remarks:

1-For nonlinear network with exponential characteristics, then the necessary and sufficient condition above for strictly increasing

nonlinearities can be applied ( result 2). Due to this condition we have : the equation  $x=w-BF(x)$  possesses a unique solution if and only if B is in  $P_0$  and  $\det(B) \neq 0$

In a first order circuit, if  $B > 0$  then B is in  $P_0$  ( by using the first property (i) of  $P_0$  matrix) and therefore there exists a unique solution. As we have mentioned in chapter 2, in practical network, matrix B in our standard equation is a non-negative matrix, this implies that the first order network with exponential characteristics always converges to a stable state ( unique solution)

In a second order circuit, the condition  $B \in P_0$  is equivalent to:

$$B_{11} \geq 0 \quad (a) \quad (\text{usually satisfied})$$

$$B_{22} \geq 0 \quad (b) \quad (\text{usually satisfied})$$

And

$$\det(B) > 0 \quad (c)$$

The above remark can be verified in the next chapter when we make an experiment with a second order circuit consisting of two p-n junctions; a unique steady state is reached whenever conditions (a),(b), and (c) are satisfied.

2-For nonlinear network with polynomial characteristics, in general there is no conclusion about the network solution. Depending on whether or not the polynomial is strictly increasing or eventually strictly increasing , the available theory on nonlinear network analysis may or may not apply.

Although recently many contributions to the issue of existence and uniqueness of nonlinear network solution have brought to a



rather complete theory to take form, there still exists many gaps in theory and points of departure for further extensions. All the studies by far were restricted to a somewhat narrow domain, namely the nonlinear functions which characterize nonlinearities in the network were restricted to either monotone increasing or eventually monotone-increasing functions. Generally applicable criteria for the determination of nonlinear network solution needs further future developments.

# Chapter 6

## Experiment and Circuit simulation results

Previous chapters have presented theoretical study of digital implementation from nonlinear analog networks. Based on a new method proposed in [1] by Dr. Frey, any nonlinear analog networks composed of two-terminal nonlinear devices can be implemented by a digital equivalent which preserves all topological properties of the original network. The new digital equivalent network contains a linear subnetwork  $L$  which is determined entirely by the original circuit topology and a nonlinear subnetwork  $N$  to solve for  $F(x)$  according to equation:

$$x(n) = w(n) - BF[x(n)] \quad (6.1)$$

Where  $x, w$  are  $k$ -vectors with  $k$  is the number of nonlinearities in the network.  $F(x)$  is a diagonal mapping from  $R^k$  into  $R^k$ .  $w(n)$  is an internal signal and is the input of subnetwork  $N$ .  $w(n)$  depends on both circuit topology and nonlinear device characteristics.

Thanks to the simplicity of standard form (6.1), we have shown that a generalized implementation of subnetwork  $N$  can be obtained for different nonlinear analog networks. This implementation also provides programmable parameters which permits the user to program subnetwork  $N$  to satisfy a specific application. Experimental results and circuit simulation of subnetwork  $N$  with its associated equation (6.1) are reported in this chapter.

## 6.1 EXPERIMENTAL RESULTS WITH EXPONENTIAL NONLINEARITIES

Making use of the exponential characteristics of a p-n junction diode, any exponential nonlinearity can be replaced by a standard diode, provided that the gain B in equation (6.1) is programmable.

The current-voltage characteristics of a p-n junction diode is defined by:

$$I=f(x)=I_{RS}(e^{x/Vt} -1) \quad (6.2)$$

Equation (6.2) can be approximated by the perfectly exponential law:

$$I=I_{RS}e^{x/Vt}$$

Where:

I=Diode conduction current

$I_{RS}$  = Reverse bias saturation current

x = Voltage across the junction

Vt is the thermal voltage,  $Vt = kT/q$

In the standard equation

$$x(t) = w(t) -BF[x(t)] \quad (6.3)$$

$I_{RS}$  can be lumped into the programmable gain B, thus with a fixed diode we can program B to satisfy any exponential nonlinearity characterized by  $f(x)=ae^{bx}$

Figure 6.1 shows an experiment circuit to solve a first order equation with exponential characteristics. In figure 6.1, DAC 0800 is a monolithic 8-bit high-speed current-output digital-to-analog converter with typical settling time of 100 nano seconds. It's used here to convert the digital input  $w(n)$  into analog input  $w(t)$ .  $b_1b_2...b_8$

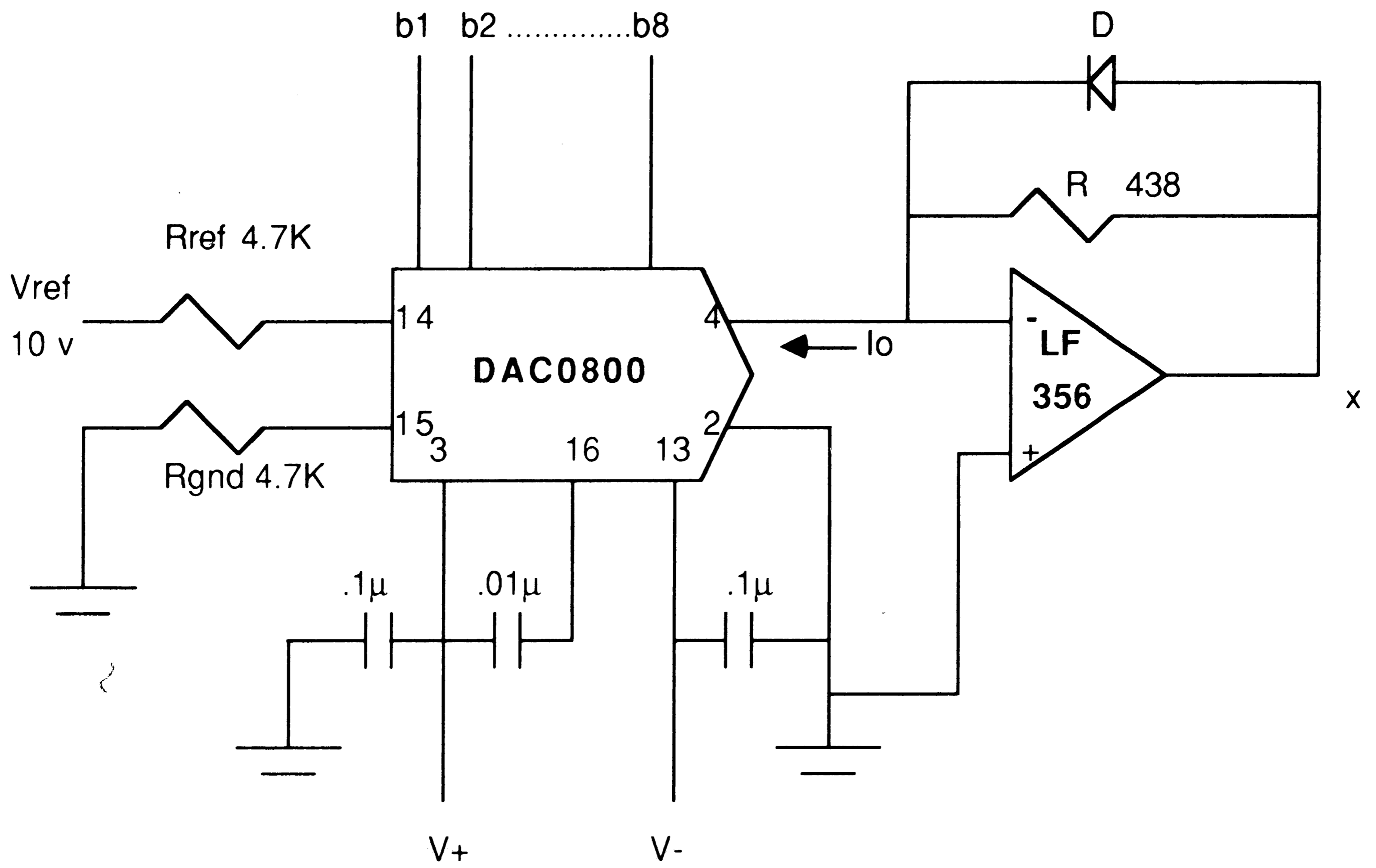


Figure 6.1

First order experiment circuit

represents the binary code of digital signal  $w(n)$  that comes from the linear subnetwork  $L$  in the associated DSP network. The circuit of figure 6.1 solves the first order equation:

$$x(t) = RI_0 - Rf[x(t)] \quad (6.4)$$

In equation (6.4),  $I_0$  is a function of binary code  $b_i$  and reference voltage  $V_{ref}$ . We can determine  $I_0$  from the diagram of DAC 0800 in figure 6.2.

From figure 6.2:

$$I_0 = I_{in} \sum_{i=1}^8 b_i 2^{-i} \quad (6.5)$$

From figure 6.1:

$$I_{in} = \frac{V_{ref}}{R_{ref}}$$

Therefore:

$$x(t) = \frac{V_{ref}}{R_{ref}} R \sum_{i=1}^8 b_i 2^{-i} - Rf[x(t)] \quad (6.6)$$

Diode 1N4148 was used as an exponential nonlinearity with the characteristics:

$$f(x) = I_{RS} e^{x/Vt}$$

At room temperature (25°C):

$$f(x) = 2.8 \cdot 10^{-14} e^{39.4x} \quad (6.7)$$

Table 6.1 lists output  $x$  measured from experiment circuit and calculated from equation (6.6) and (6.7) at different binary codes  $b_i$ 's. These values of output  $x$  are plotted in figure 6.3. Figure 6.3 also shows a transient response of output  $x$  when a pulse is applied at  $b_1$ ,

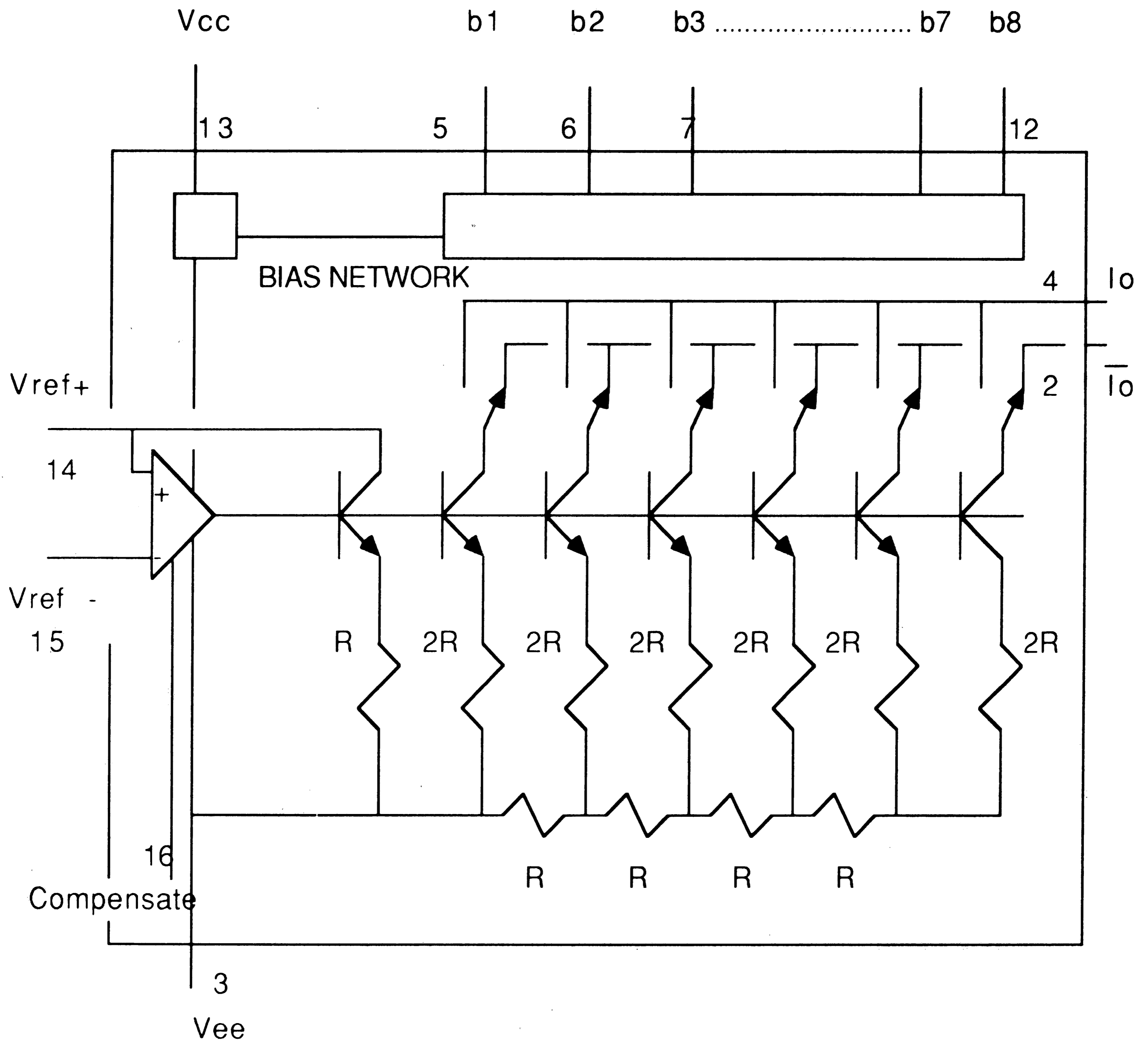


Figure 6.2

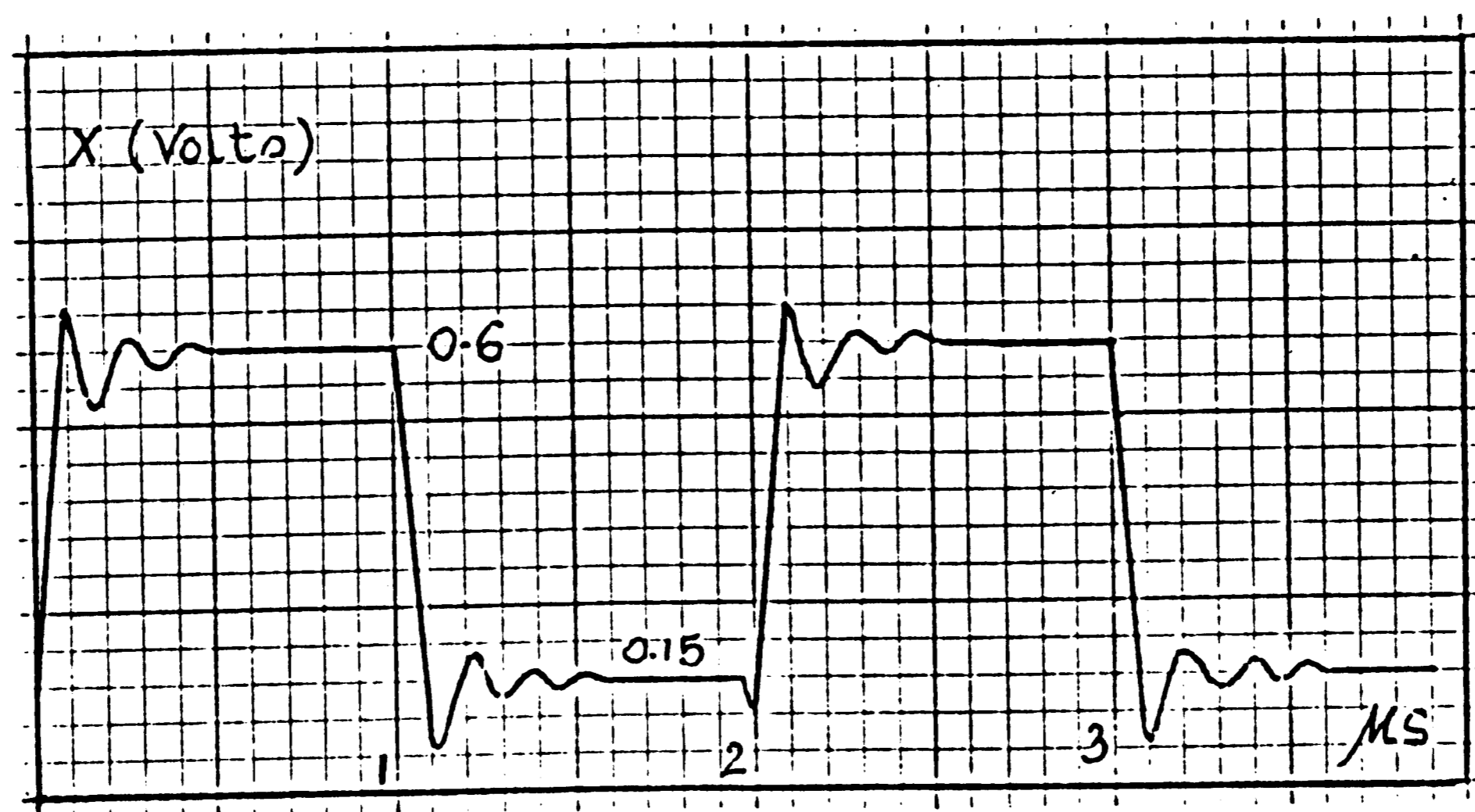
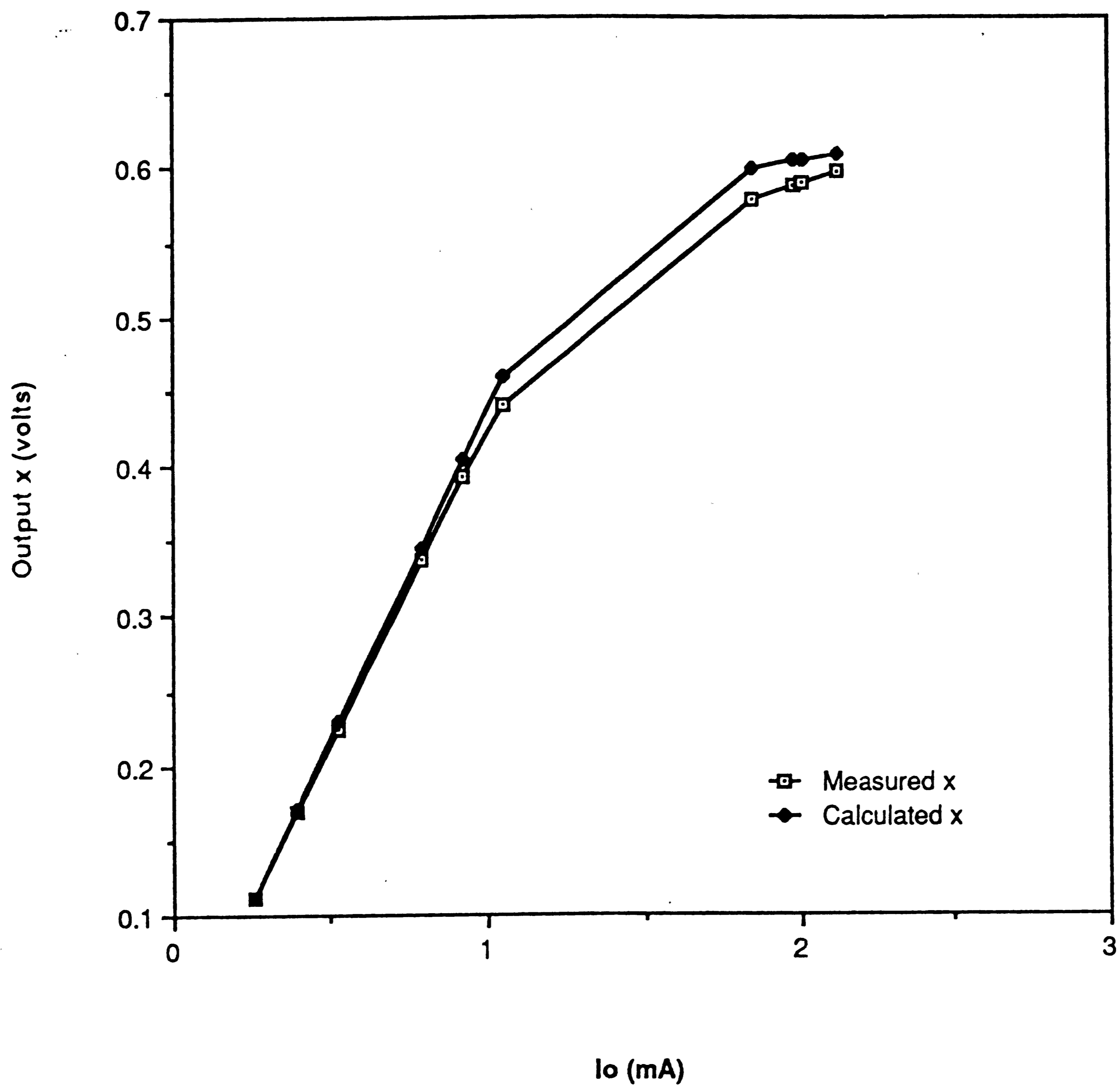
Block diagram of DAC 0800

table6.1

	b1b2...b8	Io (mA)	Measured x	Calculated x
1	00011111	0.256	0.111	0.112
2	00101111	0.390	0.169	0.171
3	00111111	0.522	0.225	0.229
4	01011111	0.786	0.337	0.344
5	01101111	0.920	0.392	0.403
6	01111111	1.050	0.440	0.459
7	11011111	1.846	0.577	0.598
8	11101111	1.980	0.588	0.604
9	11110011	2.012	0.589	0.605
10	11111111	2.120	0.597	0.609

Table 6.1

Fig.6.3a First order experiment circuit result





$b_2$  and  $b_4$  while other bits remain at logic 1. The measured settling time  $\tau$  for the first order circuit is  $0.5 \mu\text{s}$ .

A second order circuit was set up in figure 6.4. Again we used two DAC 0800's and four LF356 op-amp 's. Diode D1 is diode D above,  $I_{RS1}=2.8*10^{-14}$  Amps. D2 is a diode-connected transistor 2N3904. The measured reverse saturation current  $I_{RS2}$  of diode-connected 2N3904 is  $1.35*10^{-14}$  Amps.

Figure 6.4 corresponds to the second order equation:

$$x_1 = R_1 I_{o1} - (R_1 + R_{21}) f_1(x_1) - R_{12} f_2(x_2)$$

$$x_2 = R_2 I_{o2} - R_{21} f_1(x_1) - (R_2 + R_{12}) f_2(x_2)$$

With the component values shown and with:

$$f_1(x_1) = 2.8*10^{-14} e^{x_1/vt}$$

$$f_2(x_2) = 1.35*10^{-14} e^{x_2/Vt}, \quad 1/Vt=39.4 \text{ at } 25^\circ \text{ C.}$$

We have:

$$x_1 = 10 I_{o1} - 0.34*10^{-9} e^{39.4x_1} - 5.26*10^{-9} e^{39.4x_2} \quad (6.8)$$

$$x_2 = 10 I_{o2} - 0.062*10^{-9} e^{39.4x_1} - 5.4*10^{-9} e^{39.4x_2} \quad (6.9)$$

Where  $I_{o1}$  and  $I_{o2}$  are in mA , and :

$$I_{o1} = \frac{V_{ref1}}{R_{ref1}} 10^3 \sum_{i=1}^8 b_i 2^{-i} \quad (6.10)$$

$$I_{o2} = \frac{V_{ref2}}{R_{ref2}} 10^3 \sum_{i=1}^8 b'_i 2^{-i} \quad (6.11)$$

Varying the input codes  $b_i$  and  $b'_i$  of DAC 1 and DAC 2 and

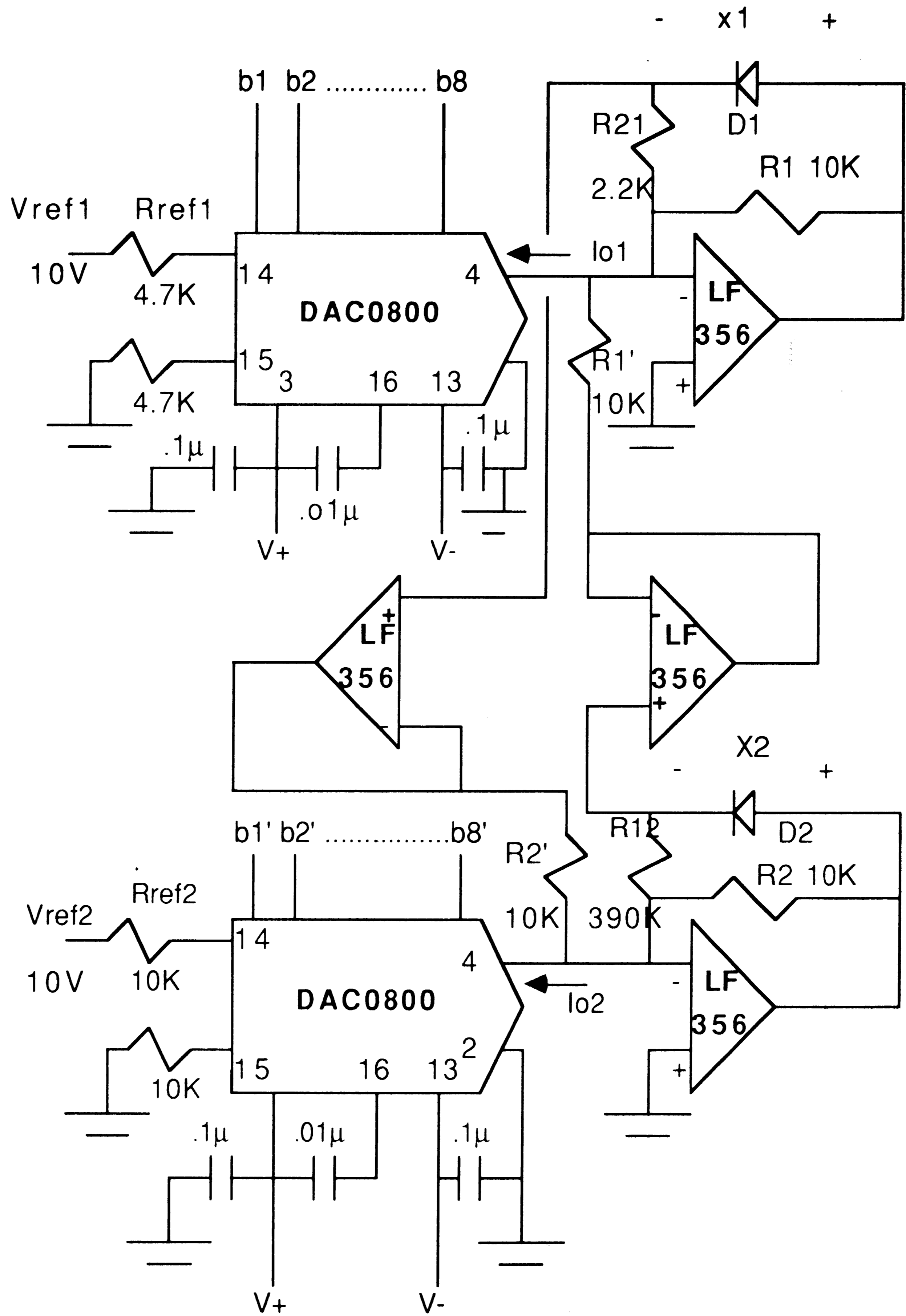


Figure 6.4  
Second order experiment circuit

measure the voltages  $x_1$  and  $x_2$  across diode D1 and D2 we have the result listed in table 6.2. Input codes  $b_i$  and  $b'_i$  are used to compute  $I_{o1}$  and  $I_{o2}$  according to equations (6.10) and (6.11). Equation (6.8) and (6.9) then are used to compute  $x_1$  and  $x_2$ . The calculated values of  $x_1$  and  $x_2$  are also listed in table 6.2 to compare with the measured values. Figure 6.5a and 6.5b show plots of  $x_1$  and  $x_2$ , including both values measured from experiment and calculated from above equations. The plots indicate a good agreement between theoretical study and practical circuit. The degree of good match between the two varies slightly from one point to another; this is due to the bulk resistance of diodes that we ignored in calculation. The transient response of  $x_1$  and  $x_2$  are plotted in figure 6.6 when a pulse is applied at bit  $b_1$  of DAC 1 while other bits remain at logic 1.

A first order equation and a second order equation with exponential nonlinearities in figure 6.7a and 6.7b were also simulated by using Spice program. These circuit employed Op-Amp LF356 with gain-bandwidth product  $GBP = 5\text{MHz}$ . In both circuits,  $w(t)$  stands for output of a digital-to-analog converter which converts digital signal  $w(n)$ . Diode D in figure 6.7a was chosen to have a typical reverse bias saturation current  $I_{RS} = 5.10^{-13}$  Amps. Diode D was used again in figure 6.7b and was named D1.  $I_{RS1} = I_{RS} = 5.10^{-13}$  Amps. Diode D2 has  $I_{RS2} = 10^{-12}$  Amps. To get a transient response of the circuit,  $w(t)$ 's are pulses which flip between two levels corresponding to two different states of binary code  $w(n)$ .

Figure 6.5a Second order experiment circuit results

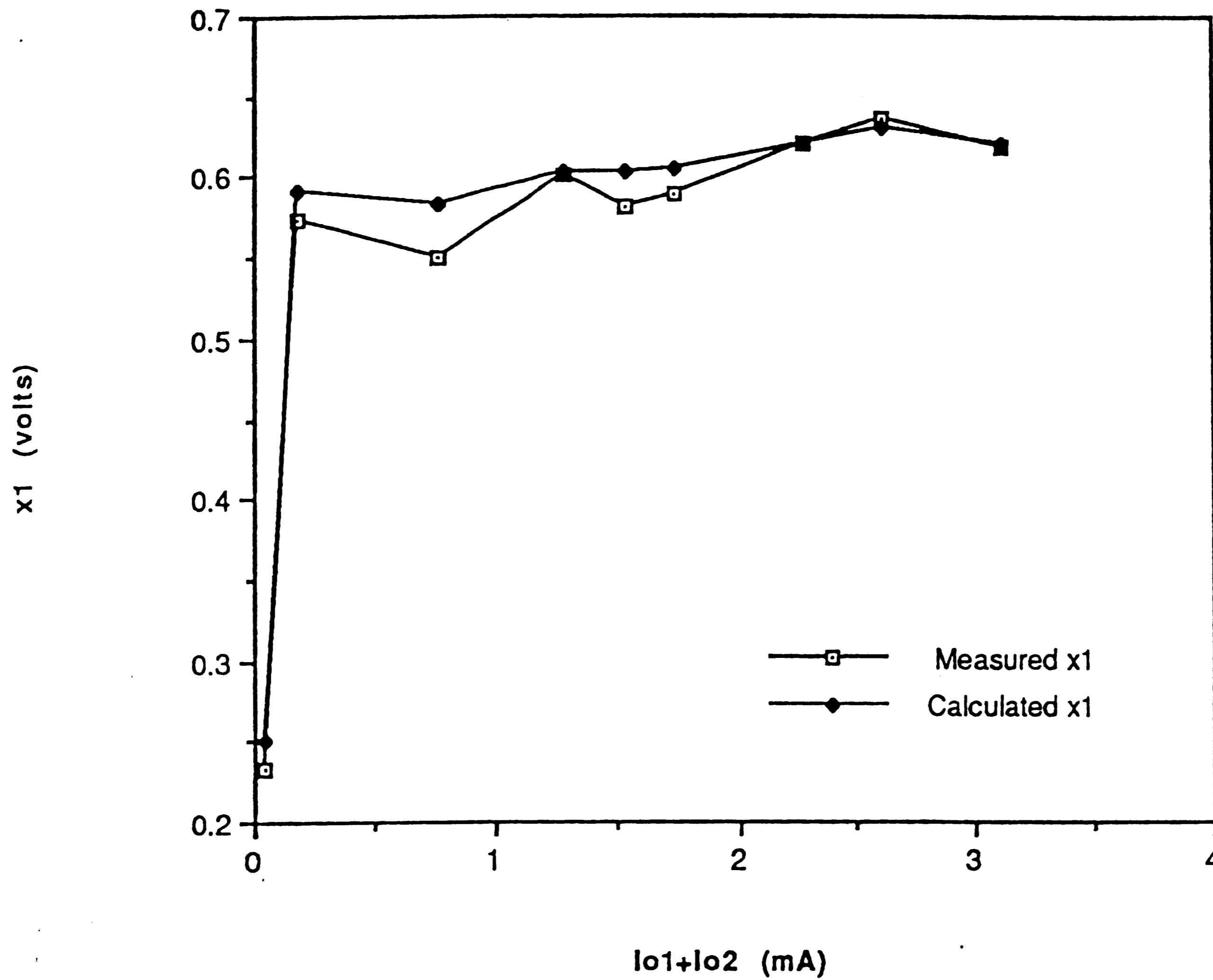


table6.2a

Thu, Apr 26,

	DAC 1 code	DAC 2 code	$I_{o1}$ (mA)	$I_{o2}$ (mA)	Measured $x_1$	Calculated $x_1$
1	00000011	00000011	0.025	0.012	0.233	0.250
2	00111111	00011111	0.125	0.058	0.573	0.592
3	00111111	00111111	0.520	0.246	0.550	0.584
4	01111111	00111111	1.050	0.246	0.600	0.602
5	01111111	01111111	1.050	0.490	0.582	0.602
6	10001111	10001111	1.190	0.558	0.590	0.605
7	11001111	10001111	1.720	0.558	0.620	0.621
8	11111111	01111111	2.120	0.490	0.636	0.629
9	11111111	11111111	2.120	0.990	0.618	0.620

Figure 6.5a Second order experiment circuit result

Figure 6.5b Second order experiment circuit results

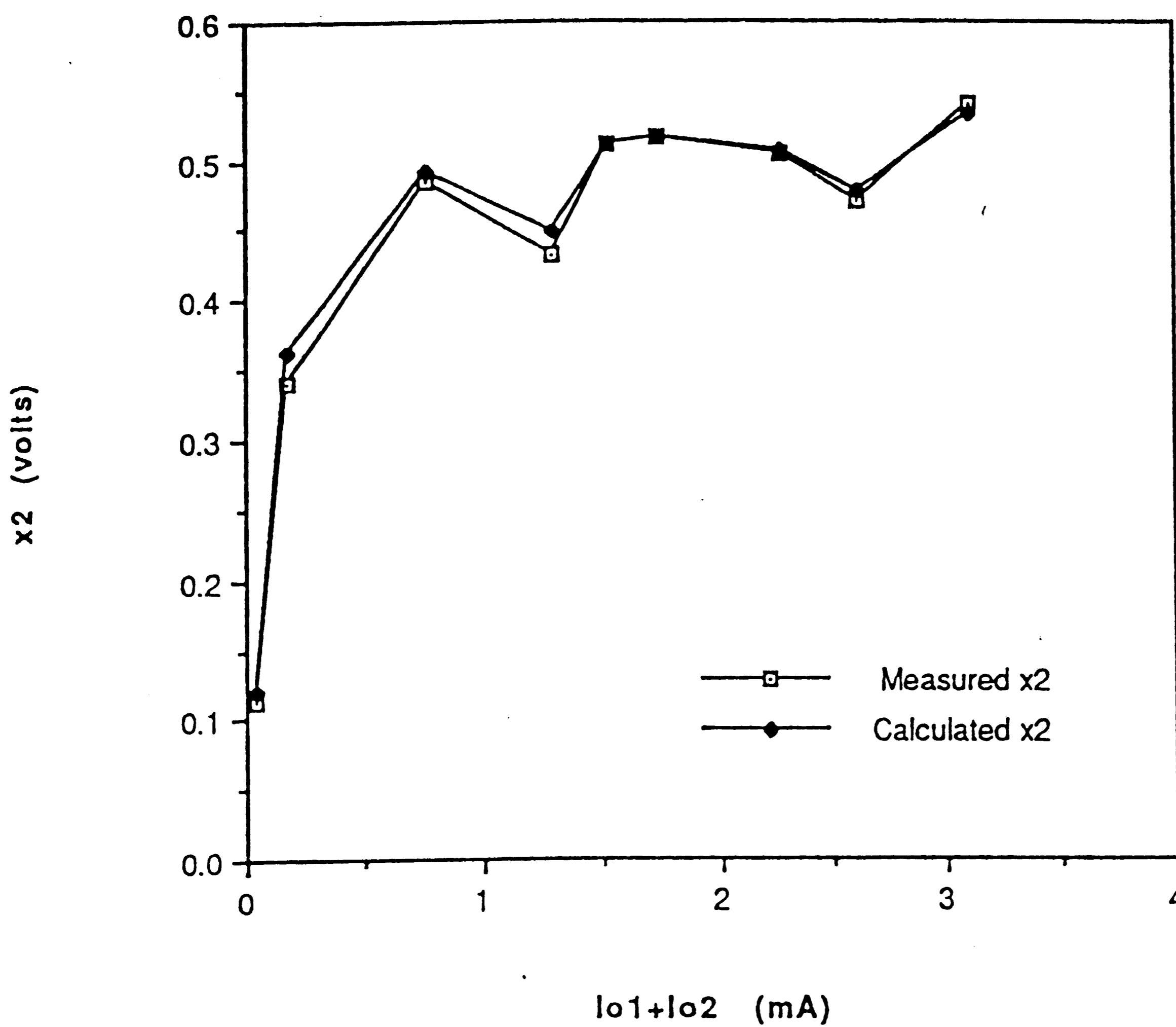


table6.2b

Thu, Apr 26,

	DAC 1 code	DAC 2 code	$I_{01}$ (mA)	$I_{02}$ (mA)	Measured $x_2$	Calculated $x_2$
1	00000011	00000011	0.025	0.012	0.114	0.120
2	00111111	00011111	0.125	0.058	0.341	0.363
3	00111111	00111111	0.520	0.246	0.484	0.491
4	01111111	00111111	1.050	0.246	0.431	0.448
5	01111111	01111111	1.050	0.490	0.512	0.512
6	10001111	10001111	1.190	0.558	0.516	0.516
7	11001111	10001111	1.720	0.558	0.504	0.506
8	11111111	01111111	2.120	0.490	0.470	0.478
9	11111111	11111111	2.120	0.990	0.539	0.532

Figure 6.5b Second order experiment circuit result

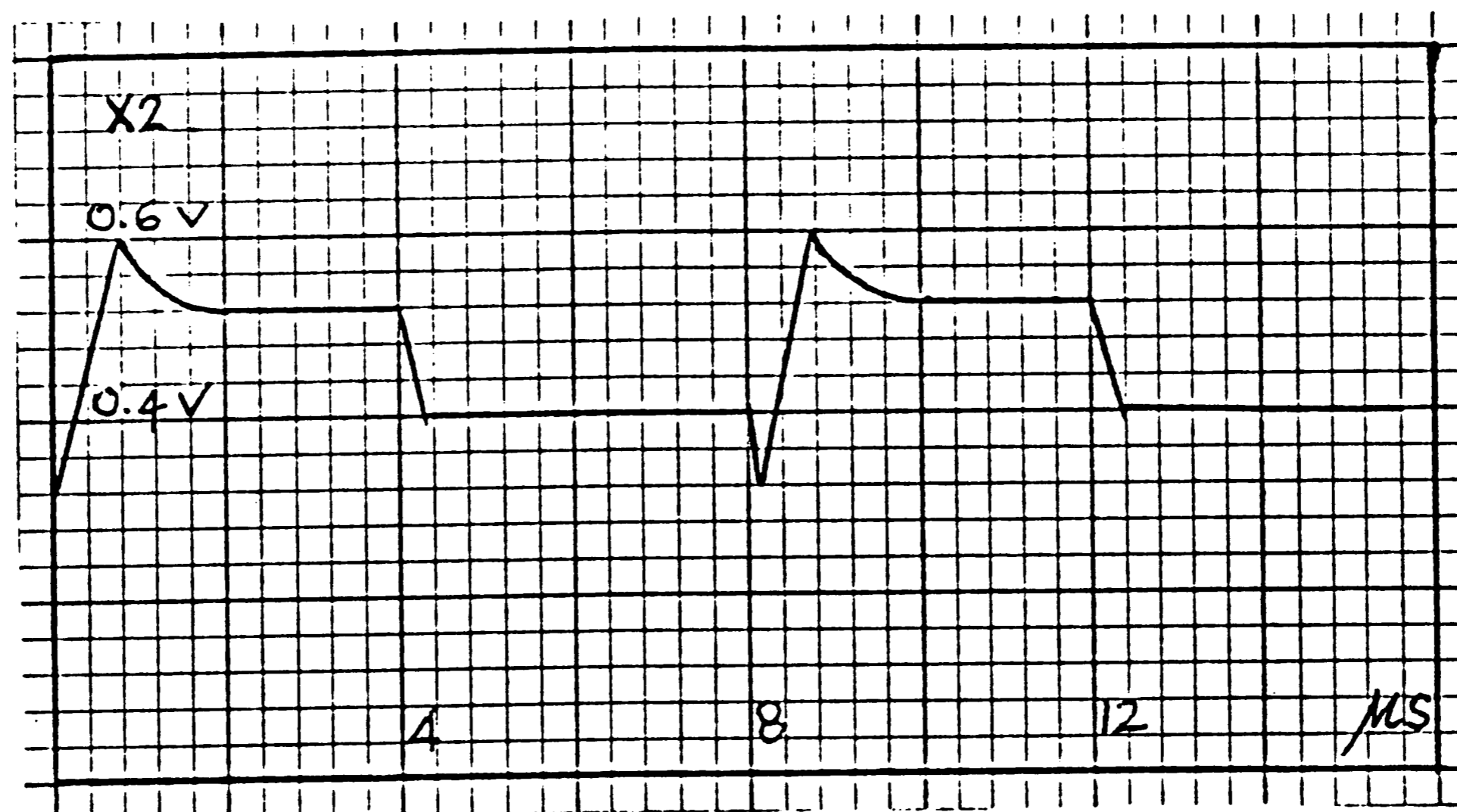
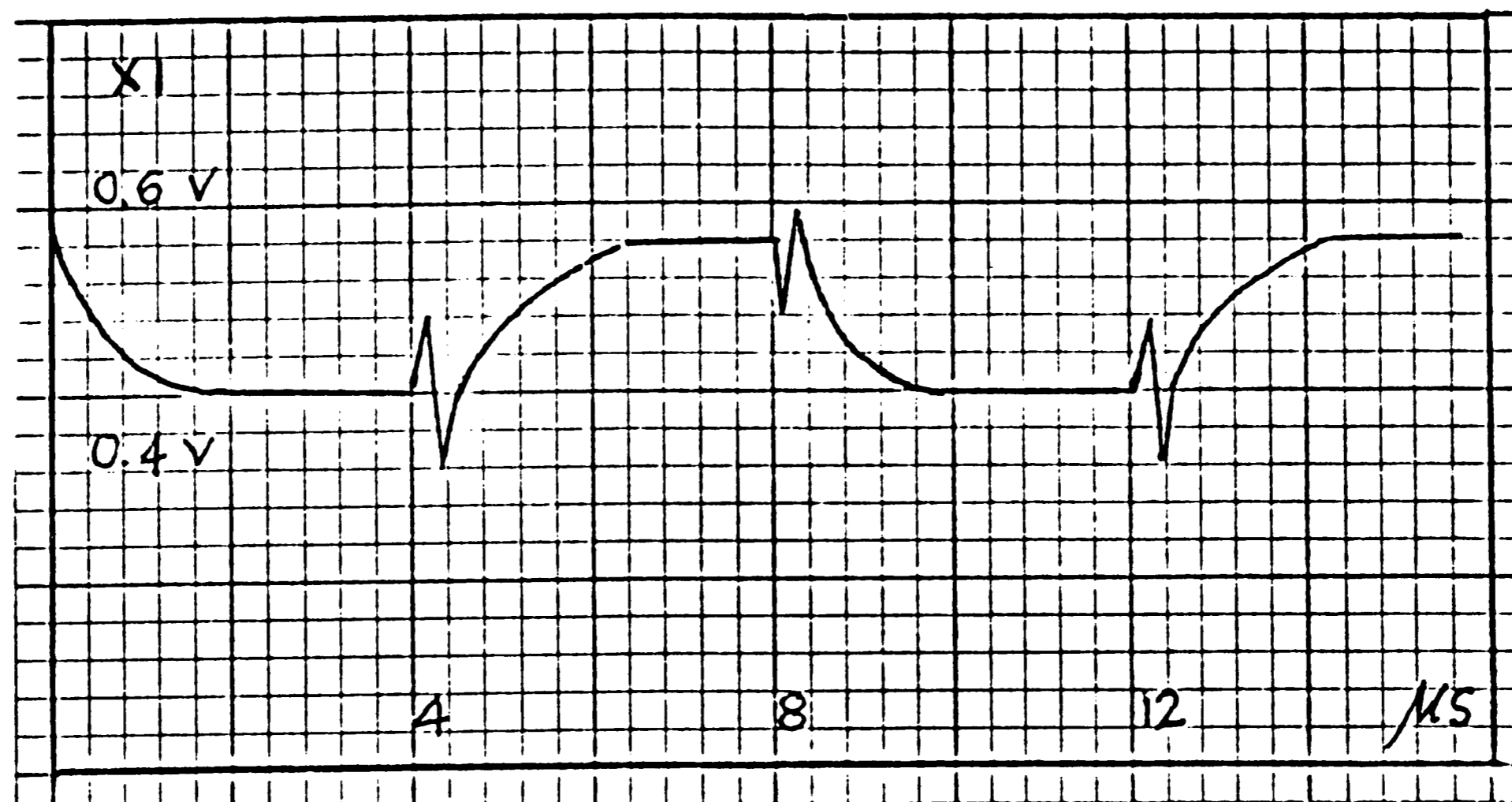


Figure 6.6

Transient response of second order circuit

With the component values shown and at the high voltage level  $w(t) = 1.5$  volts, the steady state solution of the circuit in figure 6.7a is:

$$x = 1.5 - 10^3 \cdot 5 \cdot 10^{-13} \cdot e^{x/Vt}$$

$$x = 1.5 - 5 \cdot 10^{-10} \cdot e^{x/Vt}$$

At 300 K degree  $1/Vt = 38.7 \text{ volt}^{-1}$  The solution of this equation is  $x=0.55$  volts, this result agrees with Spice result plotted in figure 6.8. The settling time is in the order of  $1\mu\text{sec}$ .

Figure 6.7b corresponds to the second order equation:

$$x1(t) = w1(t) - B11 \cdot f1[x1(t)] - B12 \cdot f2[x2(t)]$$

$$x2(t) = w2(t) - B21 \cdot f1[x1(t)] - B22 \cdot f2[x2(t)]$$

Where:

$$B11 = 1K + 2K = 3K$$

$$B22 = 398K + 2K = 400K$$

$$B12 = (B22 - R2) \cdot R1/R1' = 99.5K$$

$$B21 = (B11 - R1) \cdot R2/R2' = 2K$$

$$f1(x1) = IRS1 \cdot e^{q \cdot x1/k \cdot T} = 5 \cdot 10^{-13} \cdot e^{x1/Vt}$$

$$f2(x2) = IRS2 \cdot e^{q \cdot x2/k \cdot T} = 10^{-12} \cdot e^{x2/Vt}$$

When  $w1(t) = 2$  volts and  $w2(t) = 3$  volts, the steady state solution:

$$x1 = 2 - 1.5 \cdot 10^{-9} \cdot e^{x1/Vt} - 99.5 \cdot 10^{-9} \cdot e^{x2/Vt}$$

$$x2 = 3 - 10^{-9} \cdot e^{x1/Vt} - 4 \cdot 10^{-7} \cdot e^{x2/Vt}$$

These two equations give  $x1=0.525$  volts and  $x2=0.398$  volts.

The same results were obtained by Spice in figure 6.9. Figure 6.9a plots the input voltages  $w1(t)$ ,  $w2(t)$  and figure 6.9b plots

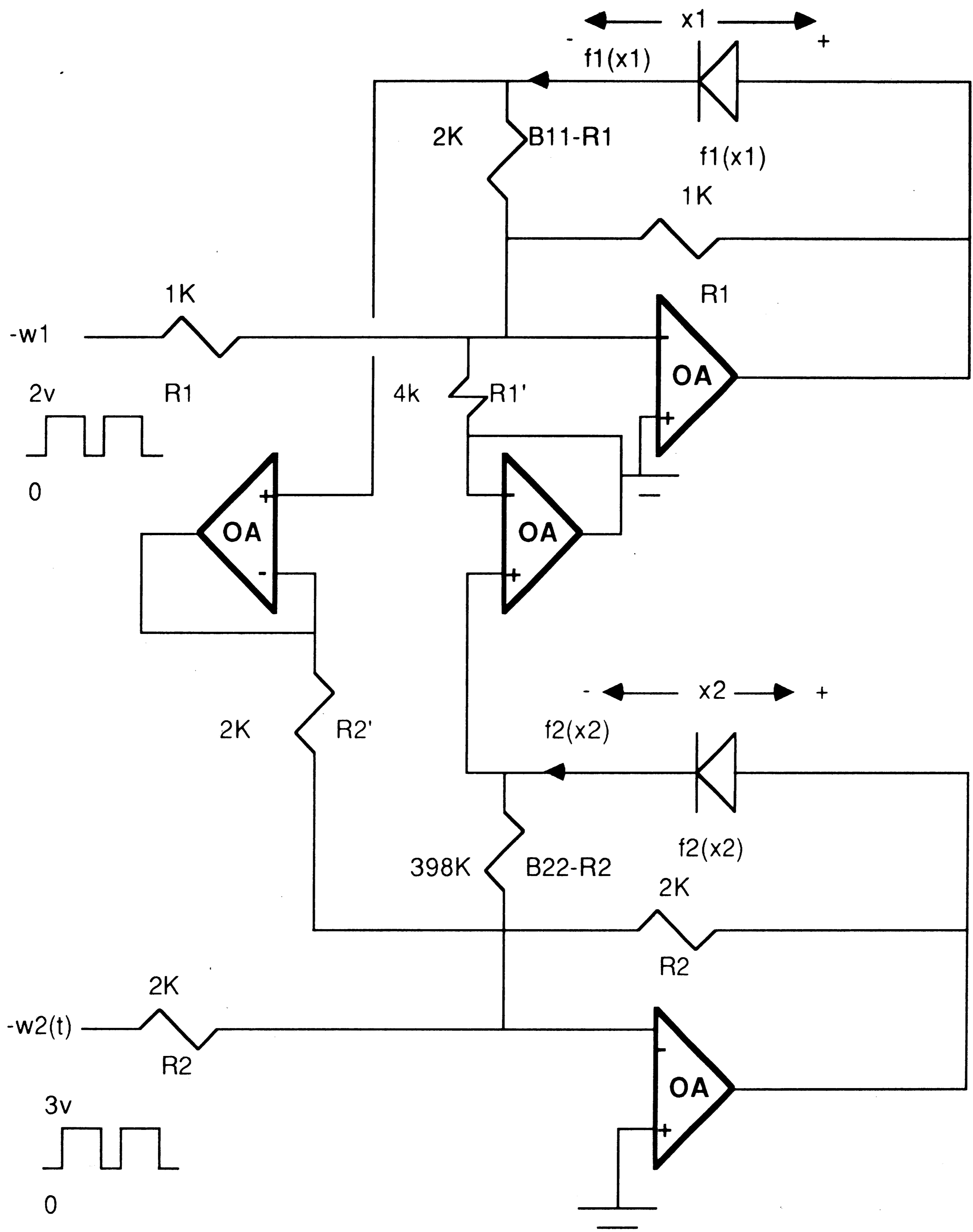


figure6.7b

Second order exponential circuit- Spice simulation



FIRST ORDER EQUATION WITH EXPONENTIAL CHARACTERISTICS-SPICE RESULT  
Date/Time run: 04/13/90 01:03.32 Temperature: 27.0

FIGURE 6.8

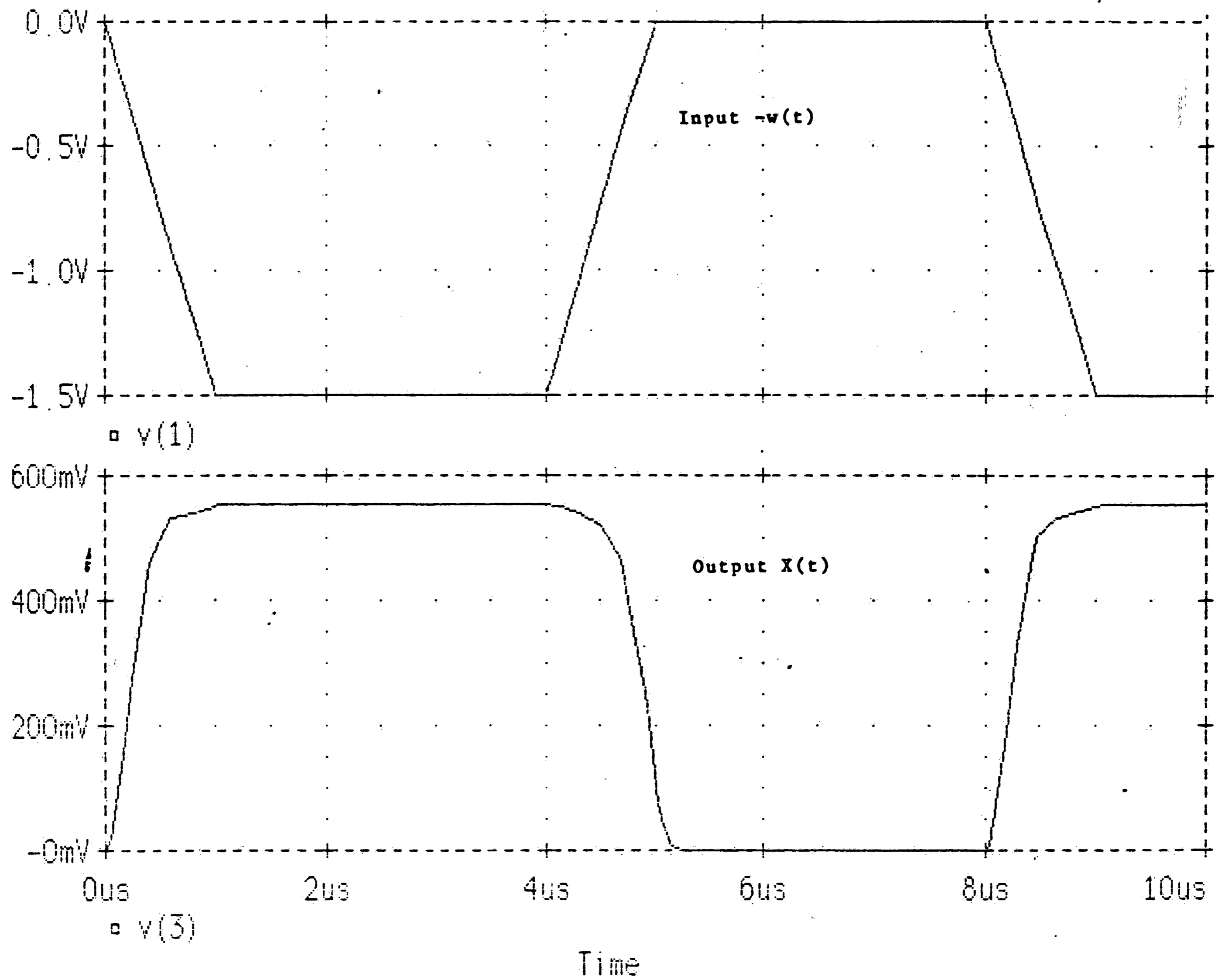


Figure 6.8

Exponential first order circuit-Spice result

SECOND ORDER EQUATION WITH EXPONENTIAL CHARACTERISTICS-SPICE RESULT  
 Date/Time run: 04/13/90 01:27:33 Temperature: 27.0

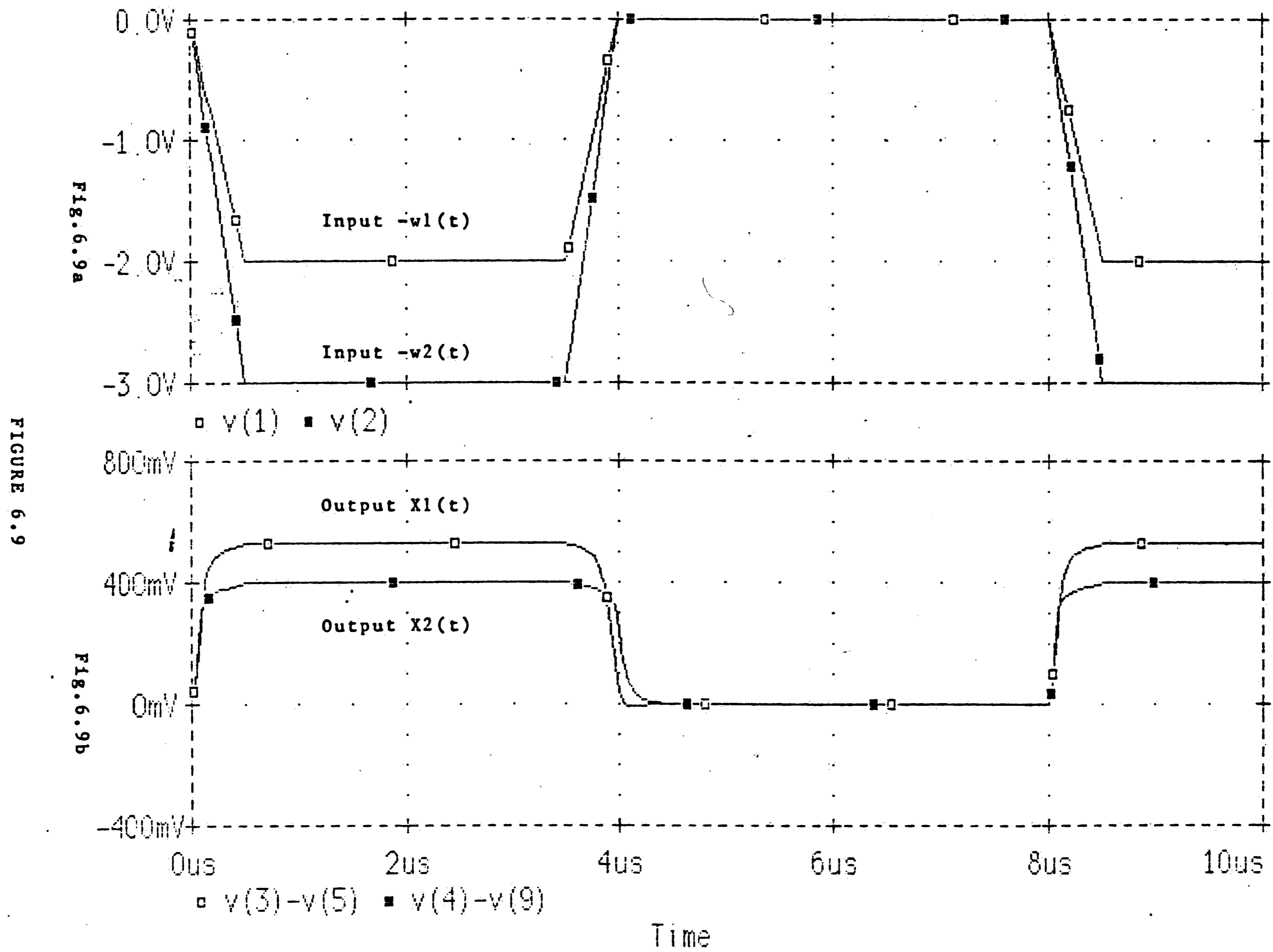


Figure 6.9

Exponential second order circuit-Spice result

the output voltages  $x_1$  and  $x_2$  across diodes D1 and D2 respectively.

## 6.2 CIRCUIT SIMULATION RESULTS WITH SQUARE LAW NONLINEAR CHARACTERISTICS

### 6.2.1 First order circuit

A first order circuit with square law characteristics is depicted in figure 6.10.

In figure 6.10, the square law function  $f(x)$  was designed to operate in an input voltage range from -10 volts to +10 volts and a maximum output current 1mA. The square law circuit and its output were plotted in figure 6.11.

Applying the results of chapter 4 to the circuit of figure 6.11a, we have:

$$I_{outmax} = \frac{2I}{n}$$

Where  $n$  is the highest order power in the circuit,  $n=2$

$$I_{outmax} = \frac{2I}{2} = I$$

Therefore the tail current  $I = I_{outmax} = 1\text{mA}$

Maximum output voltage:

$$x_{max} = R_G I$$

Gain resistor

$$R_G = \frac{x_{max}}{I}$$

$$R_G = \frac{10\text{volts}}{1\text{mA}} = 10\text{Kohm}$$

With these values of tail current  $I$  and gain resistor  $R_G$ , from

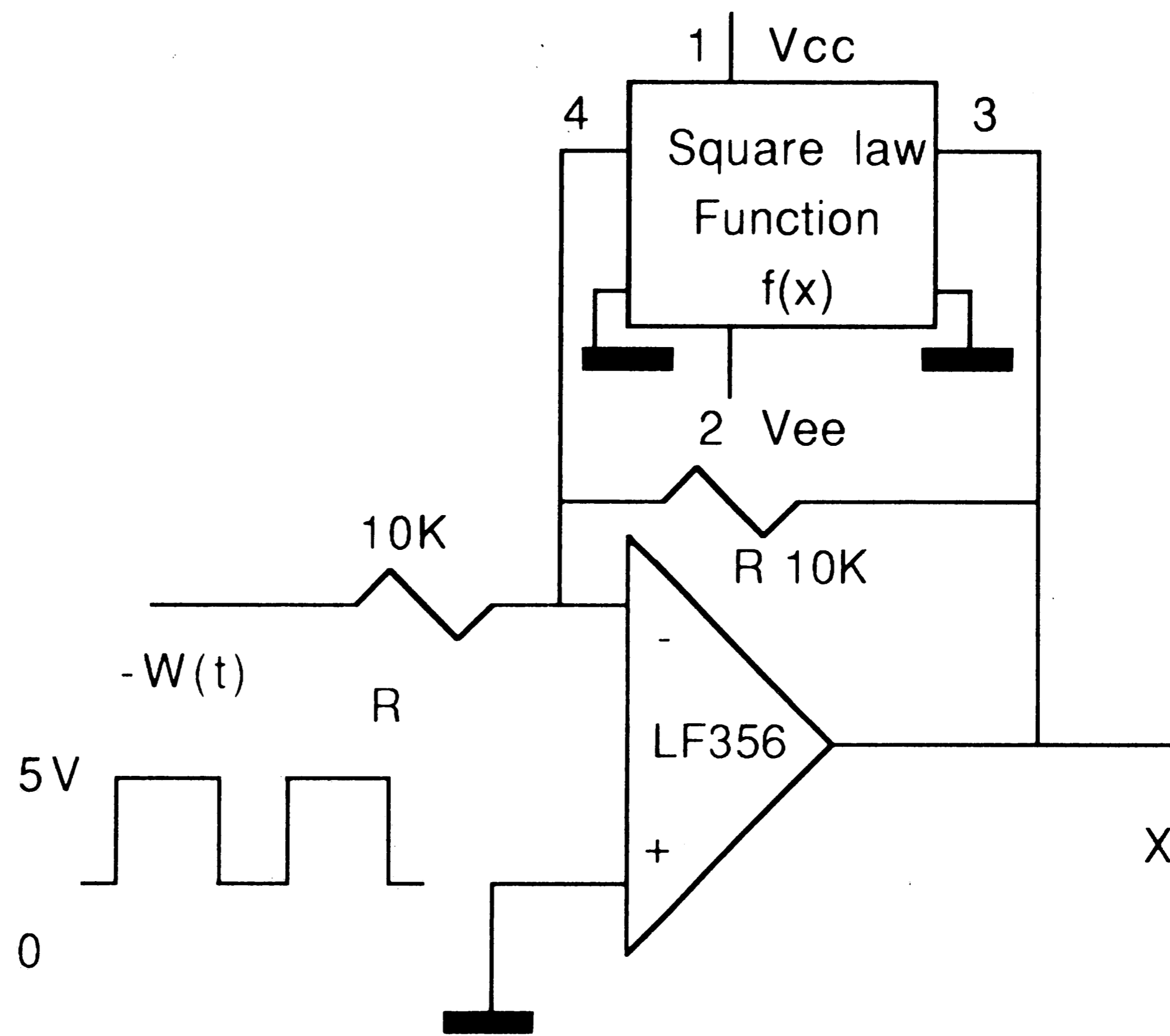


Figure 6.10

First order circuit with square law characteristics

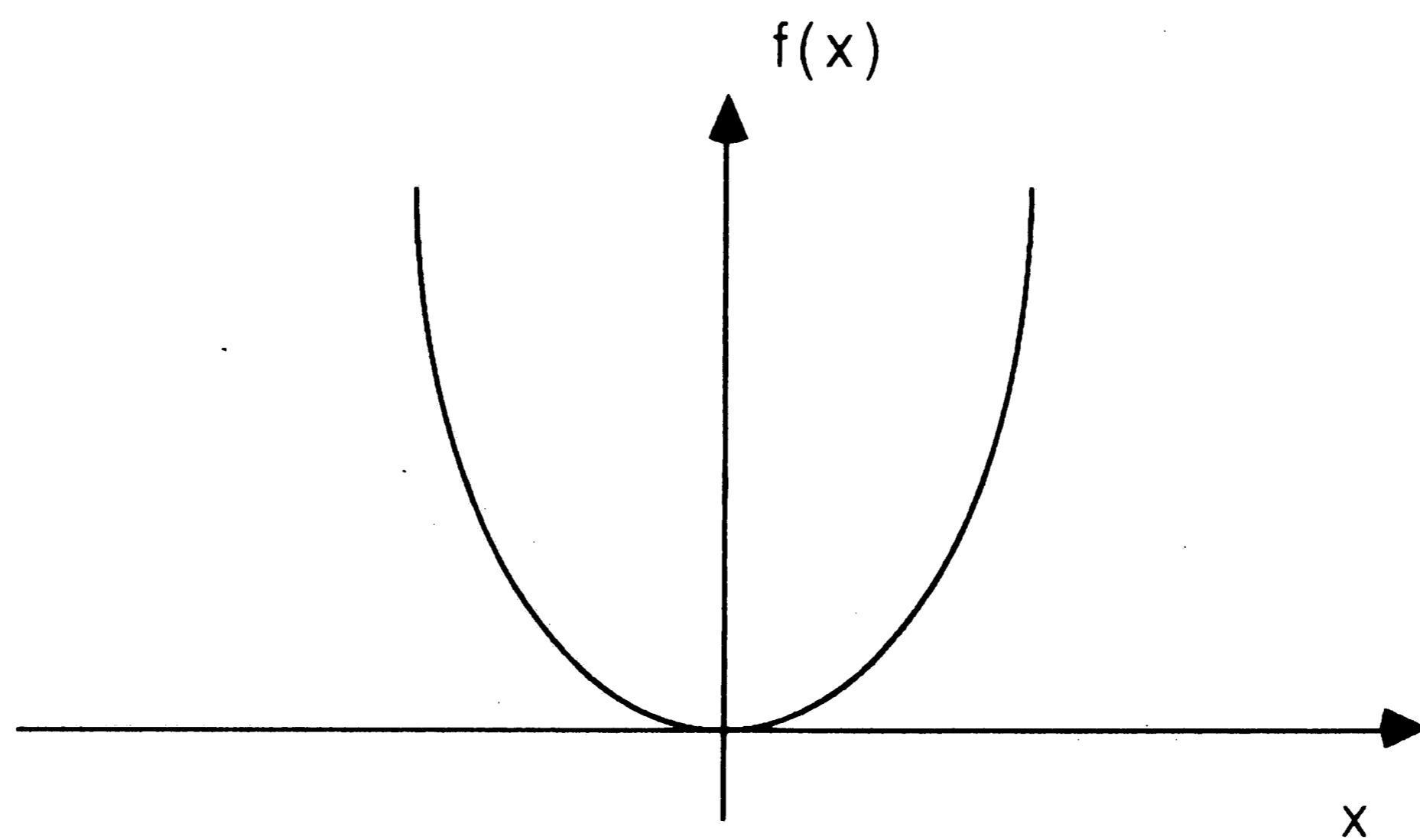


Figure 6.11b

$$f(x) = 10^{-5} x^2$$

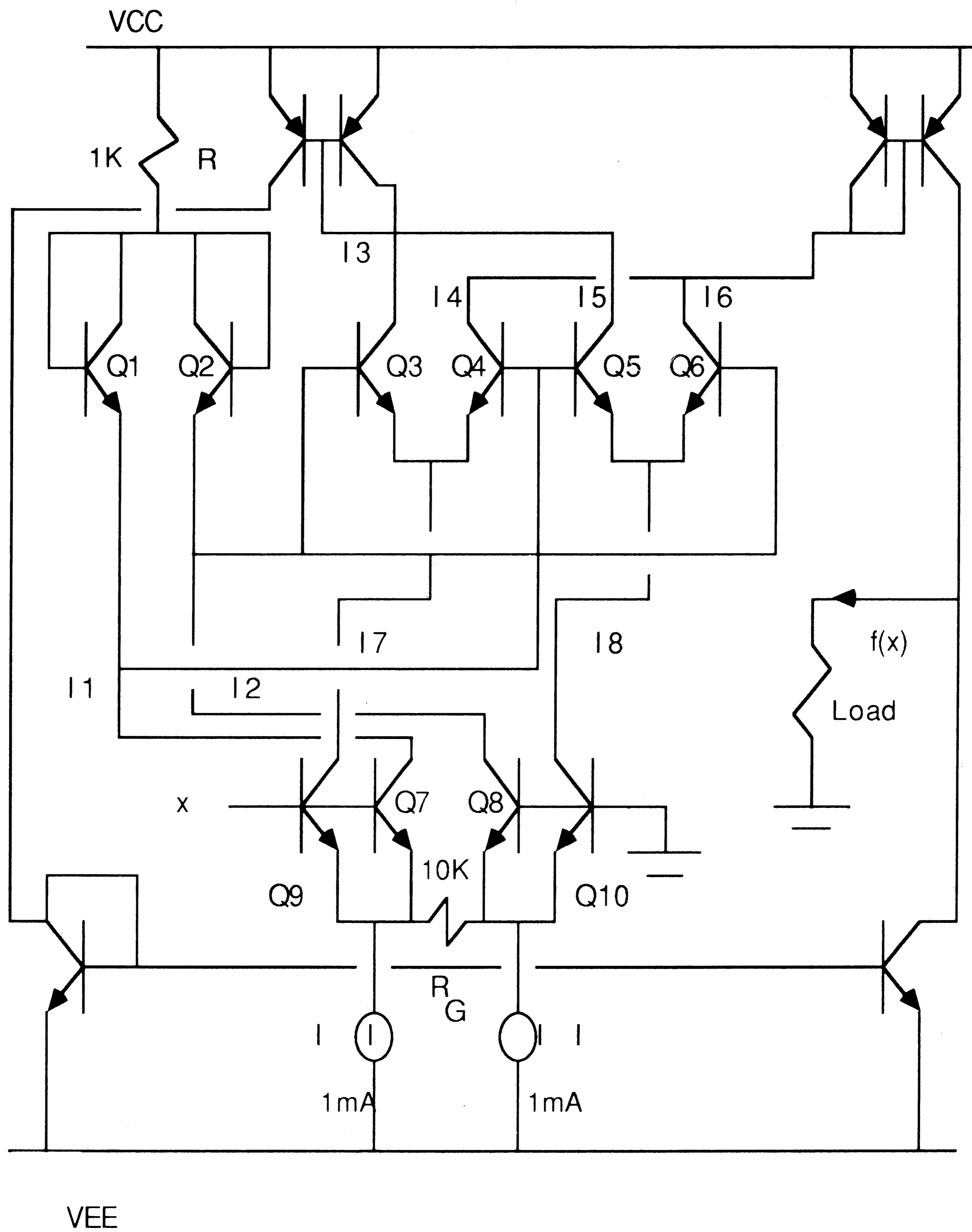


Figure 6.11a

Designed circuit to generate  $f(x) = ax^2$

equation (4.15) in chapter 4, we have:

$$I_{out}=f(x)=\frac{2x^2}{2(R_G)^2I}$$

$$I_{out}=f(x)=10^{-5}x^2 \quad (\text{Amps})$$

The first order equation associates with circuit in figure 6.10:

$$x=w(t) - Rf(x)$$

When input  $w(t)$  is at high logic level state,  $w(t) = 5$  volts, output  $x$  is the solution of equation:

$$x = 5 - 10 \cdot 10^3 \cdot 10^{-5} \cdot x^2$$

$$x = 5 - 0.1x^2$$

A positive root of this equation solved by hand is 3.66 volts, this result agrees with Spice result shown in figure 6.12. Settling time  $\tau = 1\mu\text{sec}$ .

### 6.2.2 Second order circuit

In this case:

$$x_1 = w_1 - R_{11}a_1x_1^2 - R_{12}a_2x_2^2$$

$$x_2 = w_2 - R_{21}a_1x_1^2 - R_{22}a_2x_2^2$$

Using the results of chapter 4 for two-port  $f(x)$ , we designed a second order circuit in figure 6.13. Two square law functions  $f_1(x_1)$  and  $f_2(x_2)$  were designed in the same manner with the above function  $f(x)$ .

$$f_1(x_1) = a_1x_1^2 = 2 \cdot 10^{-5}x_1^2$$

$$f_2(x_2) = a_2x_2^2 = 10^{-5}x_2^2$$

With the component values shown, we have the second order equation associate with circuit of figure 6.13:

FIRST ORDER EQUATION WITH SQUARE LAW NONLINEARITY  
Date/Time run: 04/01/90 06:49:25 Temperature: 27.0

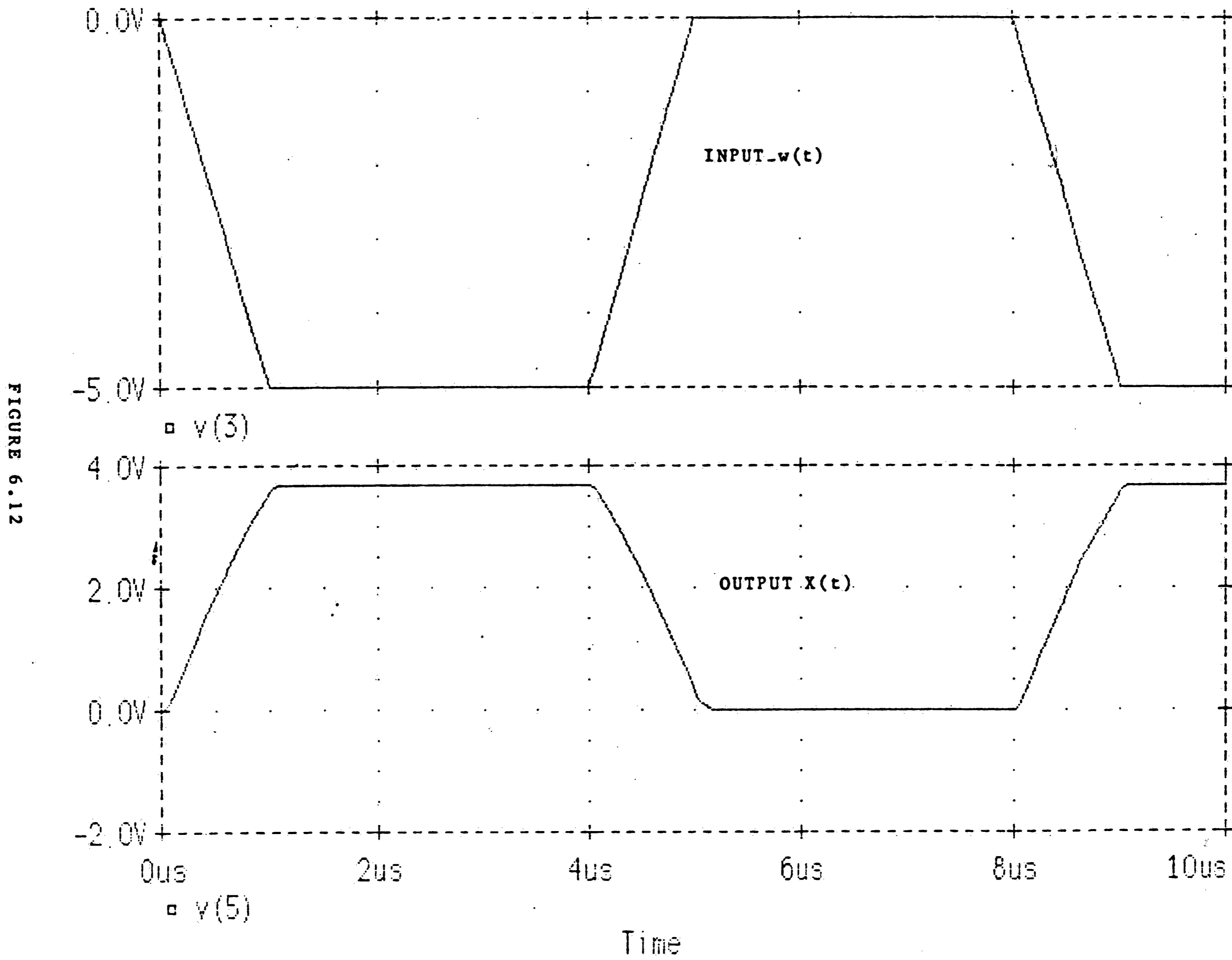


Figure 6.12

Square law first order circuit simulation result

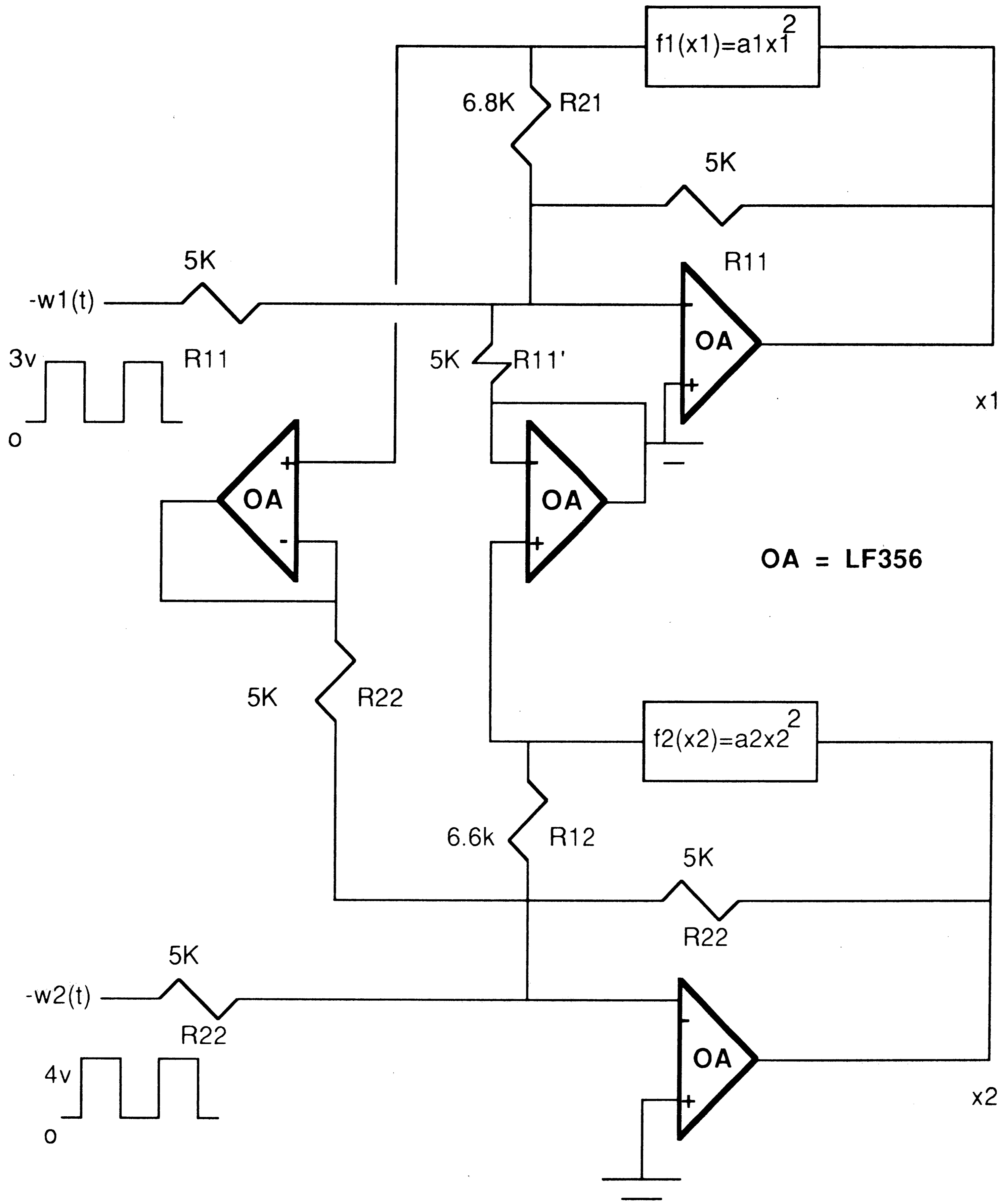


figure 6.13

Second order circuit with square law characteristics



SECOND ORDER EQUATION WITH SQUARE LAW NONLINEARITIES  
Date/Time run: 04/01/90 07:19.24 Temperature: 27.0

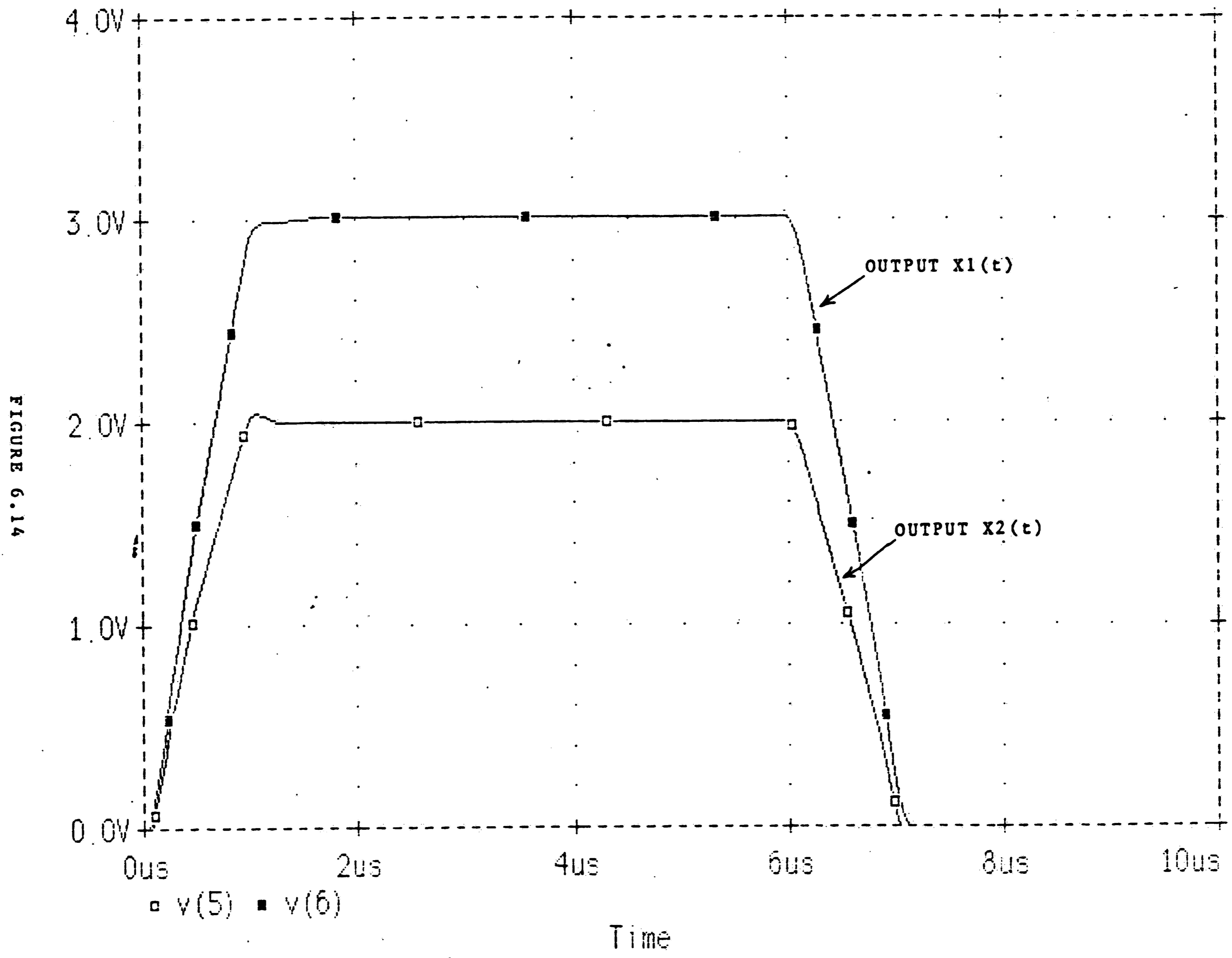


Figure 6.14

Square law second order circuit simulation result

$$x_1 = 3 - 0.1x_1^2 - 6.6 \cdot 10^{-2}x_2^2$$

$$x_2 = 4 - 13.6 \cdot 10^{-2}x_1^2 - 5 \cdot 10^{-2}x_2^2$$

Solved by hand, these two equations give positive roots at  $x_1=2$  volts and  $x_2=3$  volts. The same results were obtained by Spice in figure 6.14

### 6.3 POLYNOMIAL MODULE AND SIMULATION RESULTS

Based on circuit topology suggested in chapter 4, a third order polynomial module was designed and simulated.

The following design equations are rewritten from chapter 4:

$$\text{Maximum input voltage } x_{\max} = IR_G$$

$$\text{Maximum output current } I_{\text{outmax}} = 2I/n$$

The module was designed with  $x_{\max} = 10$  volts and  $I_{\text{outmax}} = 2.7$  mA. In a third order polynomial,  $n=3$ . From above equations we have  $I= 4$  mA and  $R_G=2.5$  Ko

The third order polynomial module with designed components is shown in figure 6.15.

The "tail" current I

$$I = \frac{V_{EE} - V_D}{R_C}$$

Resistor  $R_C$  of current source

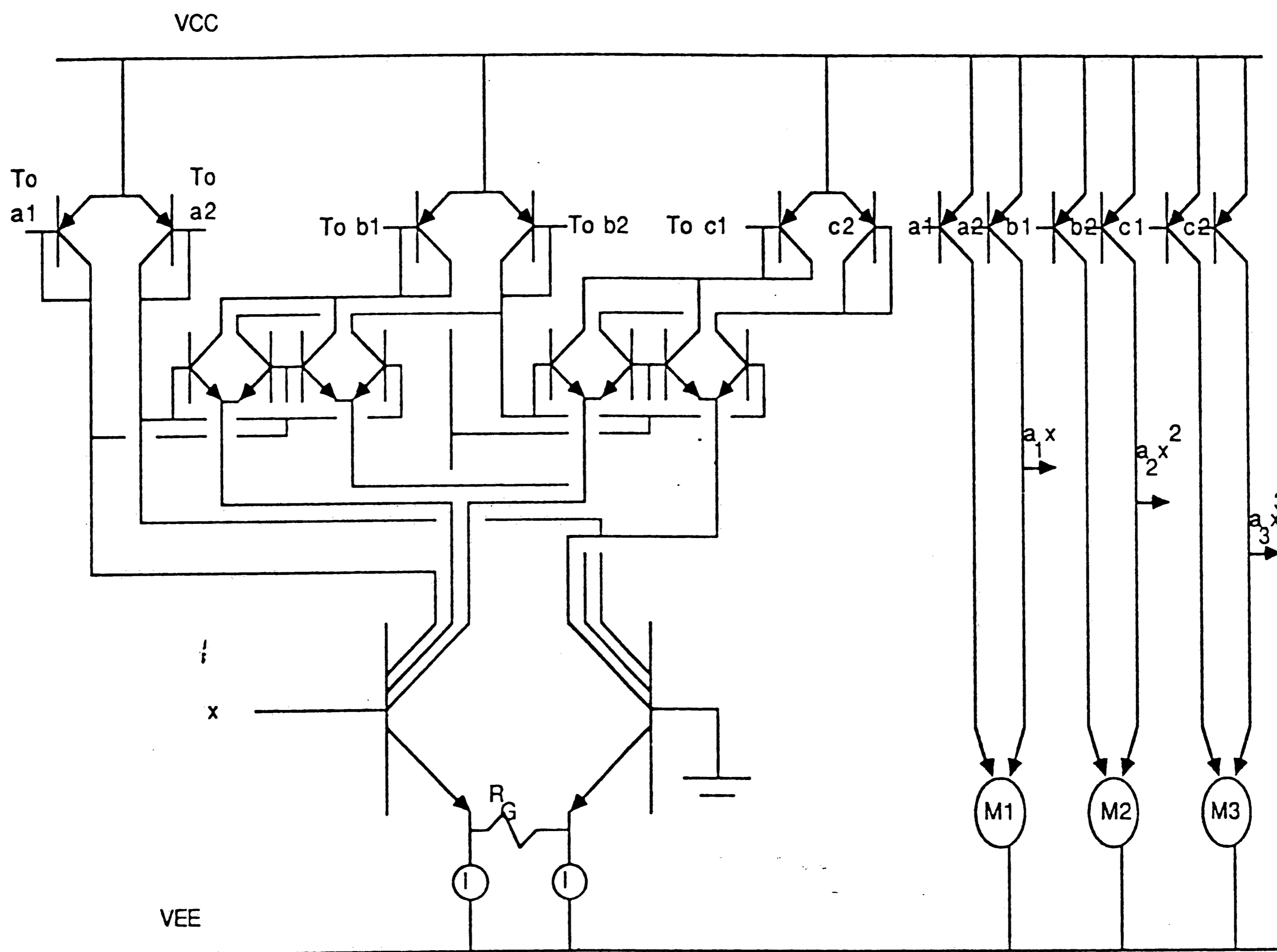


Figure 6.15

Polynomial Module simulation

$$R_C = \frac{V_{EE} - V_D}{I}$$

$$R_C = \frac{15 - 0.7}{4 \text{mA}} = 3.57 \text{Kohm}$$

Three Wilson current mirrors consist of M1, M2 and M3 were designed to avoid the off-set output current when input voltage  $x=0$

Applying equation (4.14) and (4.15) of chapter 4,

$$f_k(x) = a_k x^k$$

$$a_k = \frac{2}{n(R_G)^k I^{k-1}}$$

With  $n=3$   $R_G = 2.5 \text{ Ko}$ ,  $I=4 \text{ mA}$  we have:

$$a_1 = \frac{2}{3 \times 2.5 \times 10^3} = 2.7 \times 10^{-4}$$

$$a_2 = \frac{2}{3(2.5 \times 10^3)^2 4 \times 10^{-3}} = 2.66 \times 10^{-5}$$

$$a_3 = \frac{2}{3(2.5 \times 10^3)^3 (4 \times 10^{-3})^2} = 0.3 \times 10^{-5}$$

Thus three outputs of the module are:

$$f_1(x) = a_1 x = 27 \times 10^{-5} x \quad (\text{Amps}) \quad (\text{i})$$

$$f_2(x) = a_2 x^2 = 2.66 \times 10^{-5} x^2 \quad (\text{Amps}) \quad (\text{ii})$$

$$f_3(x) = a_3 x^3 = 0.3 \times 10^{-5} x^3 \quad (\text{Amps}) \quad (\text{iii})$$

The polynomial output is the summation:

$$f(x) = \sum_{i=1}^3 a_i f_i(x)$$

Figure 6.16, 6.17, and 6.18 show three outputs  $f_1(x)$ ,  $f_2(x)$ , and  $f_3(x)$  of the module. The curves with solid dot are ideal curve corresponding to three functions (i), (ii), and (iii). With high  $\beta$

POLYNOMIAL CIRCUIT

Date/Time run: 04/18/90 17:38:49

Temperature: 27.0

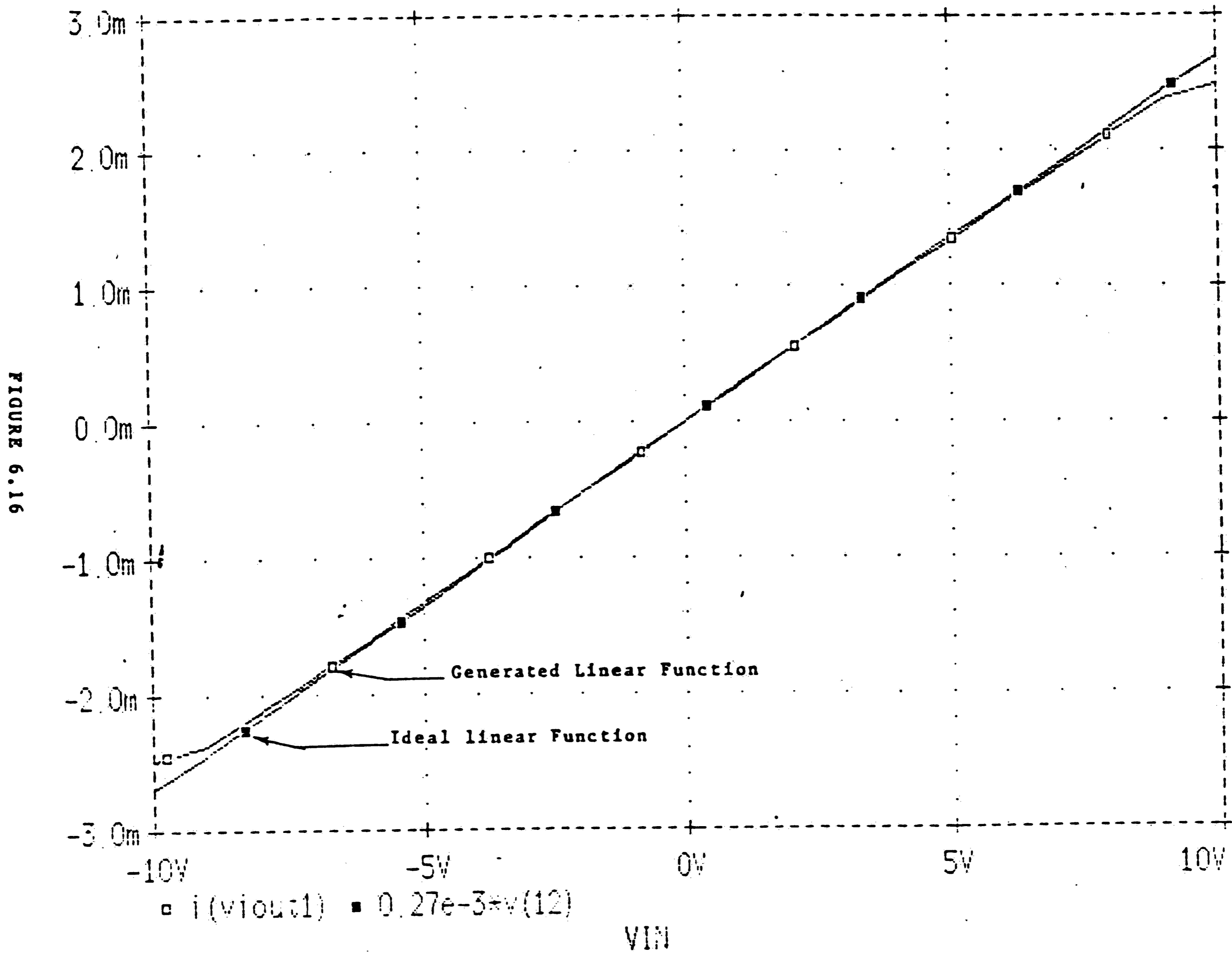


Figure 6.16

Polynomial circuit simulation result

POLYNOMIAL CIRCUIT

Date/Time run: 03/17/90 09:14:22

Temperature: 27.0

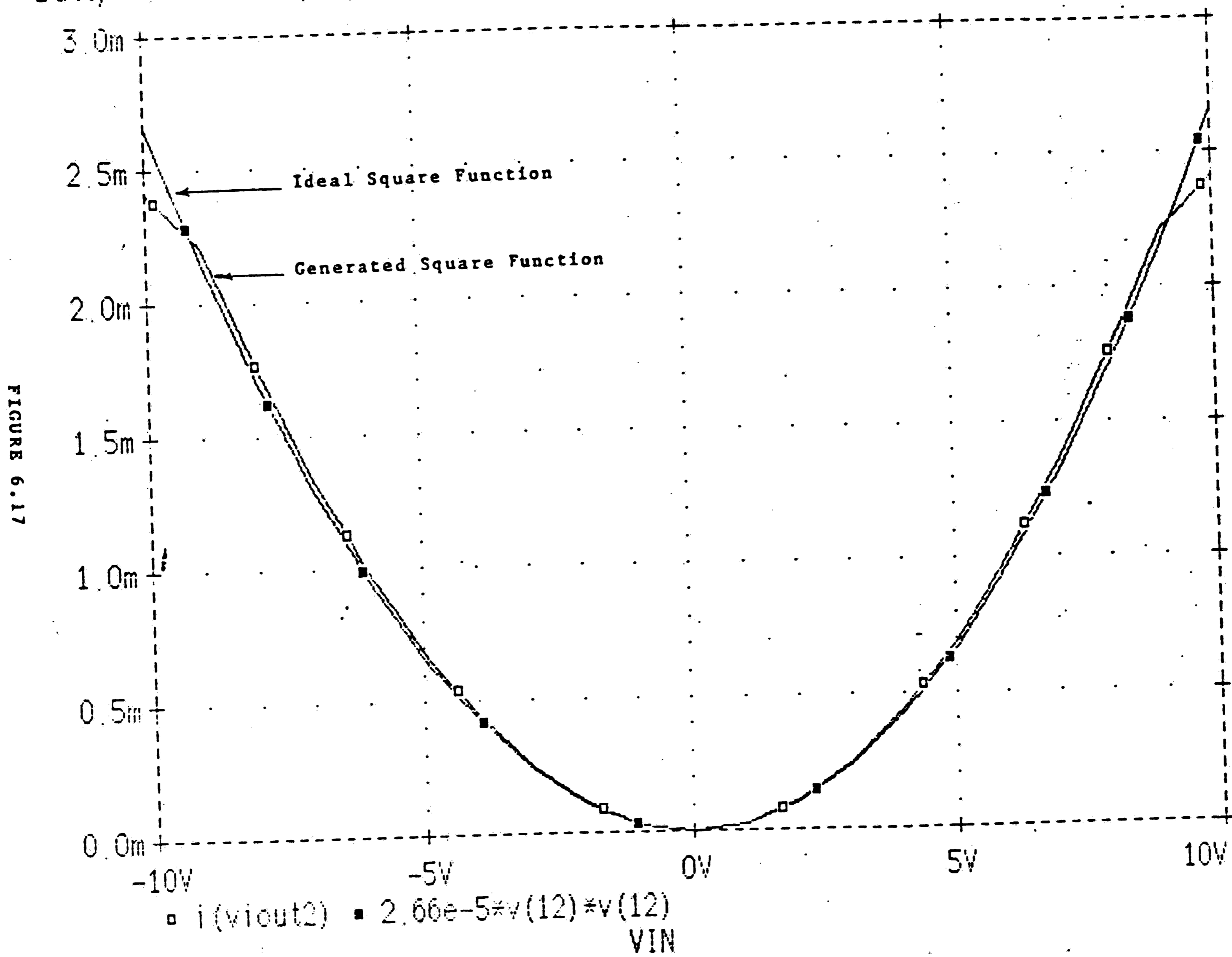


Figure 6.17

Polynomial circuit simulation result

POLYNOMIAL CIRCUIT

Date/Time run: 03/17/90 09:14:22

Temperature: 27.0

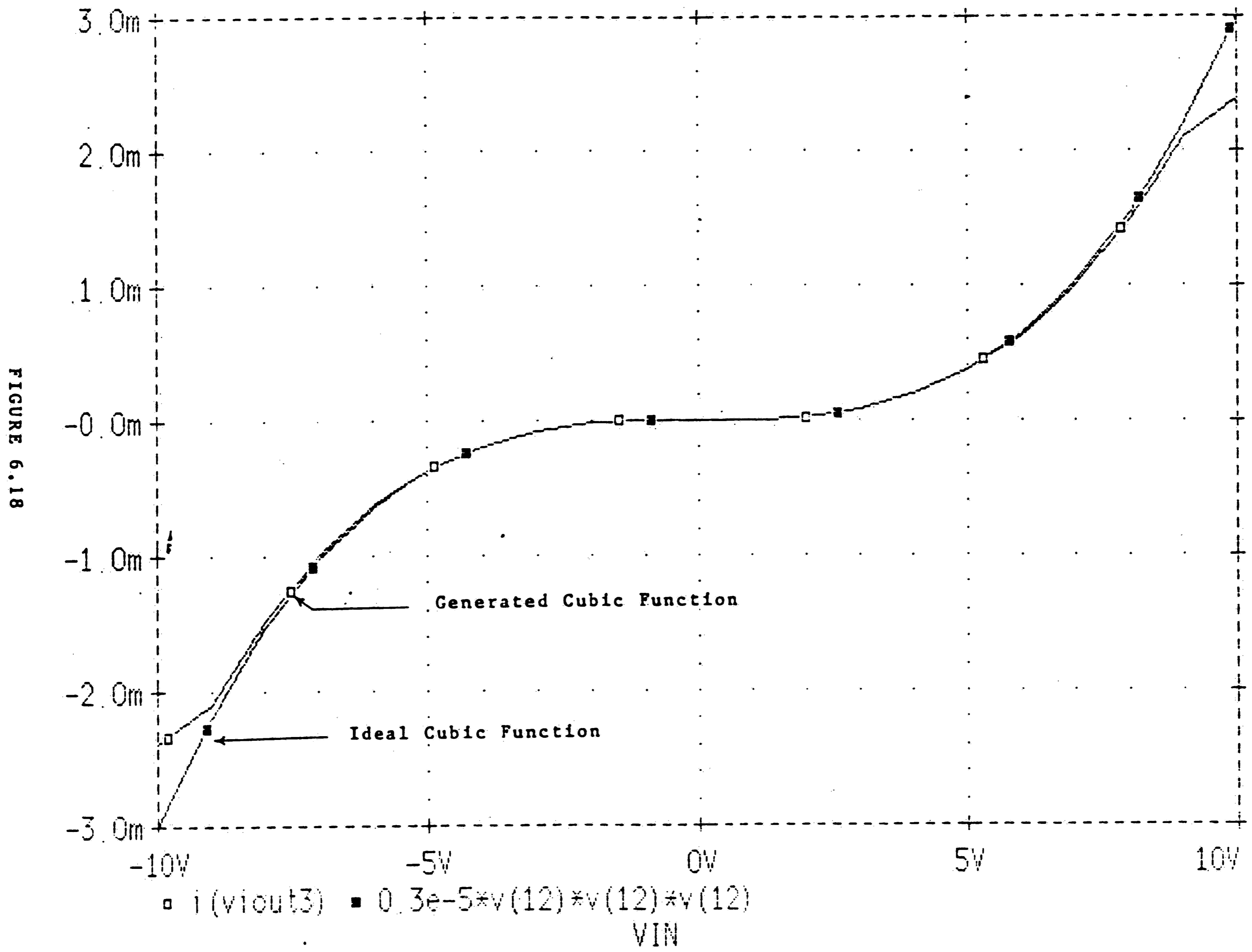


Figure 6.18

Polynomial circuit simulation result

transistors, we can achieve very accurate polynomial function at the output of our module.

## **6.4 POLYNOMIAL WITH PROGRAMMABLE COEFFICIENTS**

Virtually, every differentiable function can be replaced by an equivalent polynomial. A polynomial module can be used to simulate all nonlinear devices if its coefficients are programmable. In chapter 4 we have shown how we can use a current programming circuit to program the coefficients of the module. This section reports circuit simulation results of programmable polynomial module (PPM)

A typical circuit of a PPM is shown in figure 6.19. In figure 6.19, the current programming circuit is basically a current amplifier with the current gain  $k$  is controlled by a programming code  $b_i$ . The square law circuit is the same circuit in previous section. At the output of current programming circuit we have a square law function  $f(x) = kax^2$ , where  $k$  is a function of binary code  $b_i$  and varies between -1 and +1. By varying binary code  $b_i$  we can change the coefficient of second order term ; other terms in the polynomial can be changed by the same manner.

The current programming circuit was designed in figure 6.20. It contains a multiplying R-2R ladder providing binary multiplier; the remaining circuit provide switch implementation. Transistors Q1- Q7 are current sources supplying steering-current to BJT



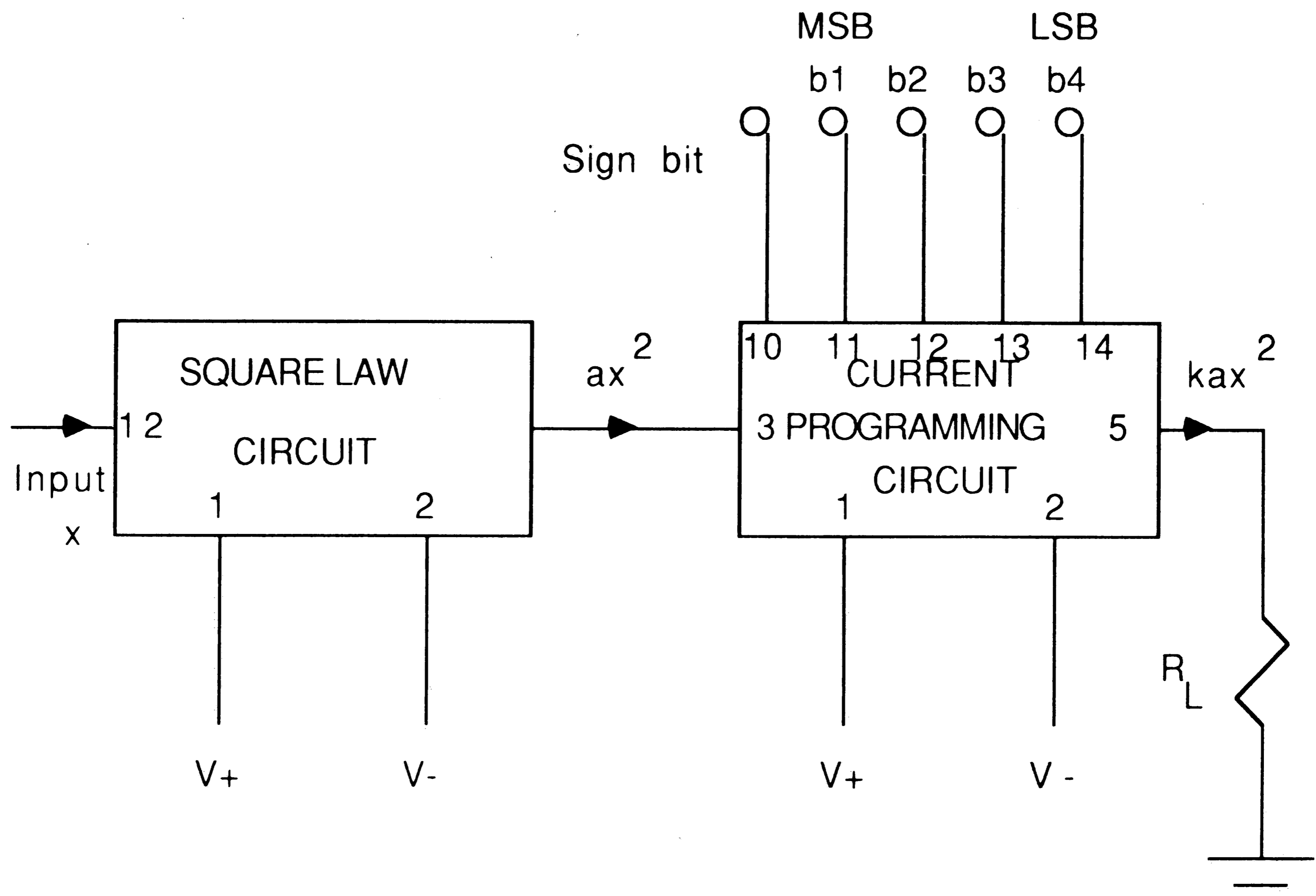


Figure 6.19

Programmable Polynomial circuit

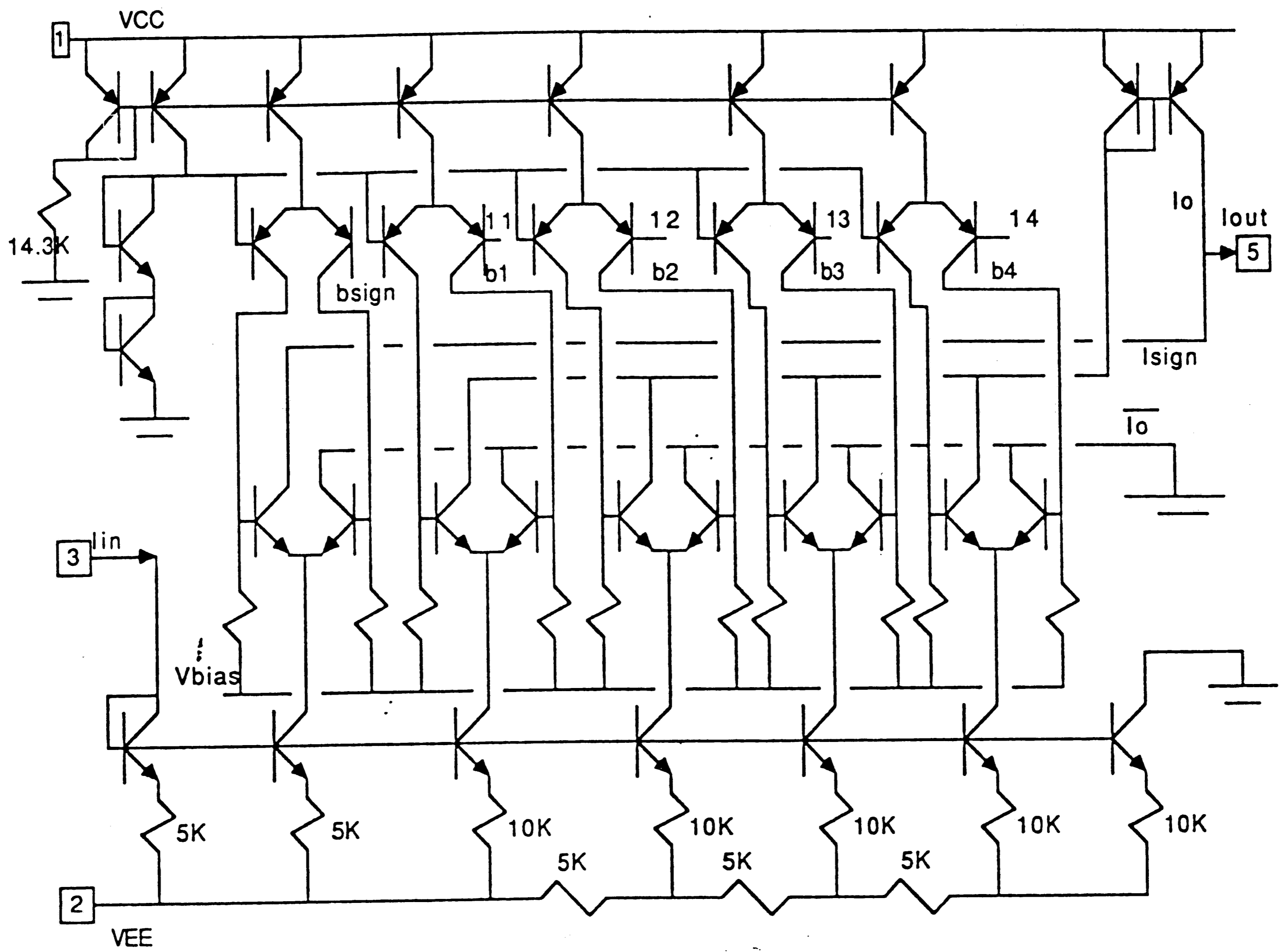


Figure 6.20

Current programming circuit

POLYNOMIAL FUNCTION CIRCUIT WITH PROGRAMMING COEFFICIENTS

Date/Time run: 02/25/90 01:22:22

Temperature: 27.0

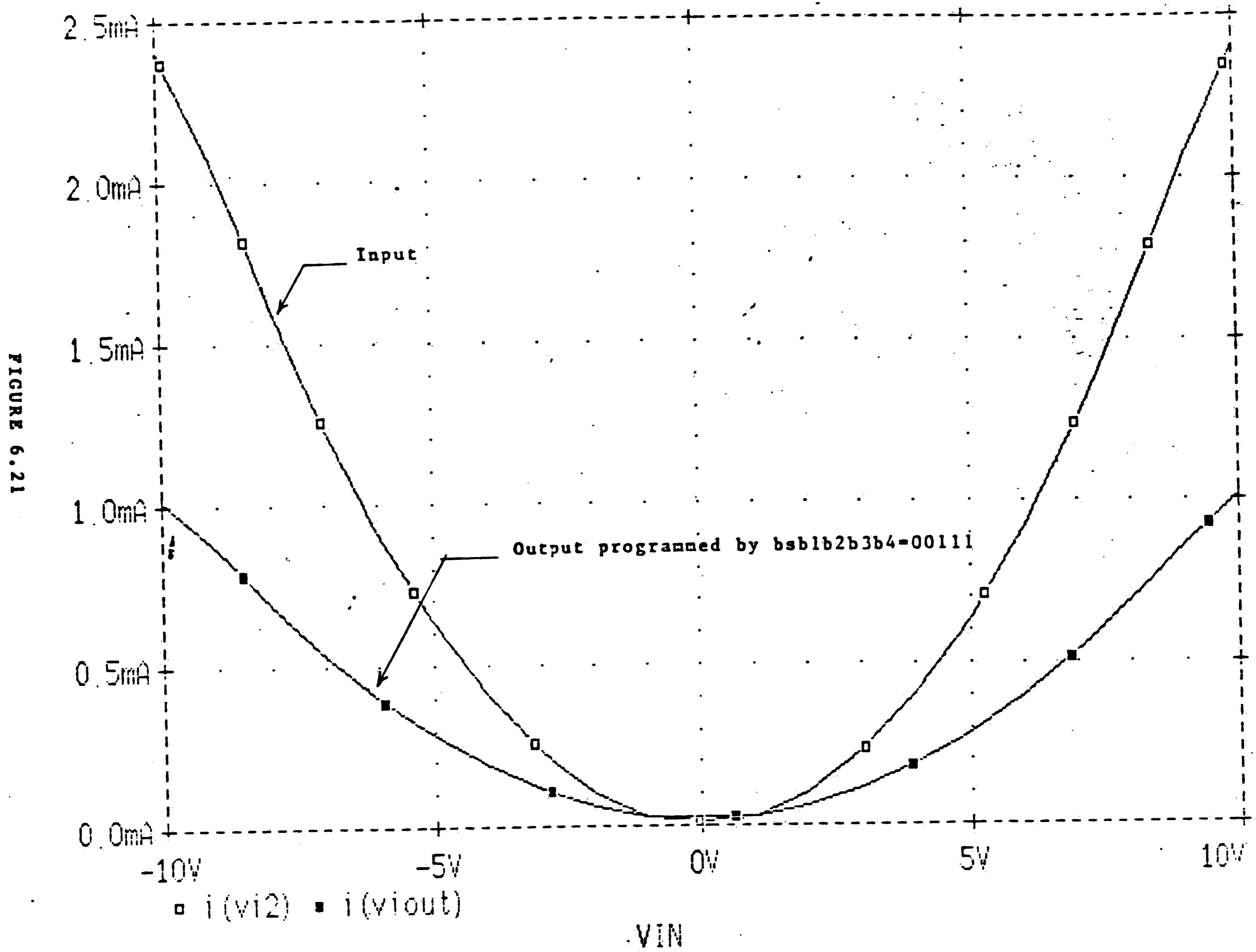


Figure 6.21

Polynomial coefficients programming

POLYNOMIAL FUNCTION CIRCUIT WITH PROGRAMMING COEFFICIENTS

Date/Time run: 02/25/90 01:37:53

Temperature: 27.0

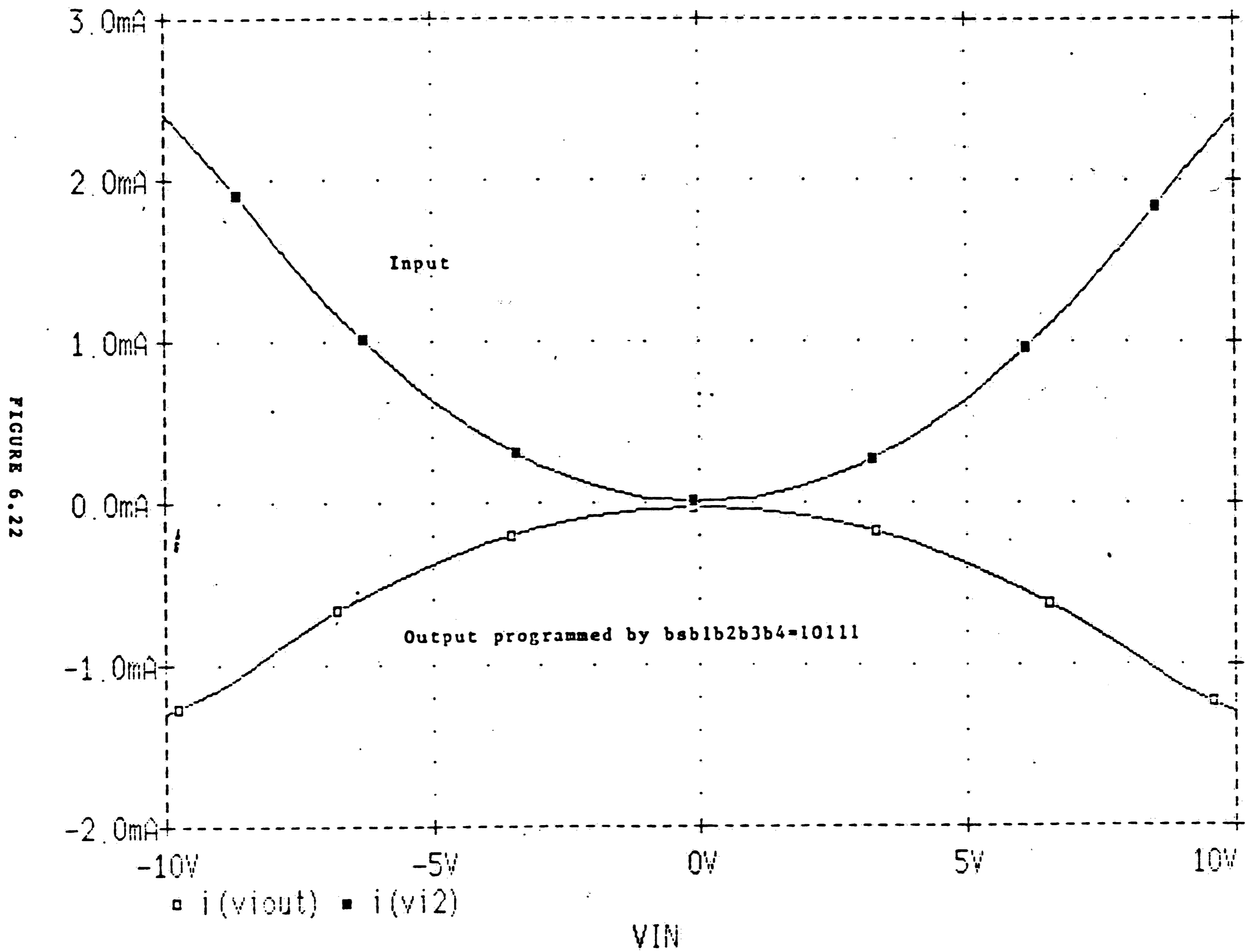


Figure 6.22

Polynomial coefficients programming

switches. Being biased in non-saturating steering-current mode, these BJT switches work very rapidly.

The current output  $f(x)$  of figure 6.19 was programmed by two different programming codes:  $b_{\text{sign}}b_1b_2b_3b_4 = 00111$  and  $B_{\text{sign}}b_1b_2b_3b_4=10111$ . Two correspondent outputs  $f(x)$  are plotted in figure 6.21 and 6.22 together with the initial output of the square law circuit. This simulation results show that by using current programming circuits we can program all coefficients of the polynomial module.

## Chapter 7

# Summary and conclusion

In the past few decades the understanding of nonlinear networks has become more advanced and their application are beginning to be appreciated. As a result of increasing applications, nonlinear network has found itself particularly interested in digital signal processing area. Nonlinear discrete-time signal processing techniques are being promoted by some fields of application. A notable example is in the area of telecommunication where nonlinear discrete-time signal processing techniques, microelectronic technology and fiber optic transmission combine to change the nature of communication system in truly revolutionary ways. A similar impact can be expected in many other areas of technology. Unfortunately, so far there is no available technique to implement a digital equivalent network from its nonlinear analog counterpart. Such a technique is obviously desired and could change the whole picture of nonlinear discrete-time signal processing area.

Recently it has been shown by professor Frey in several papers of his work [1] [8] that by using a new modeling technique we can define an associated linear network from any nonlinear analog structure containing two terminal nonlinear devices. This associated network allows the determination of equations in the standard form:

$$\mathbf{x} = \mathbf{A}\mathbf{u} + \mathbf{B}\mathbf{F}(\mathbf{x}) \quad (7.1)$$

Where  $\mathbf{x}$  is an  $n$ -vector composed of tree branch voltages and link currents, and is controlling variable of nonlinear devices;  $n$  equals

the number of nonlinearities in the network. A and B are linear operators determined by network topology, and  $F(x)$  is a diagonal mapping from  $R^n$  into  $R^n$ .

The standard form of equation (7.1) is of fundamental importance, it gives a good deal of insight to the nature of network structure and stability, it is the key to solve for any variable in the original network; and more importantly, it is the key to implement a digital equivalent one.

The digital equivalent network which is studied in this thesis comprises of linear systems characterized by impulse responses  $a(n)$  and  $b(n)$  and a solve-for- $F(x)$  subnetwork N.  $a(n)$  and  $b(n)$  can be immediately defined once we obtain the associated network by replacing all nonlinear devices by their models. The problem of deriving the digital equivalent network now reduces to the problem of implementation for the subnetwork N. We have shown that this subnetwork N receives an internal signal  $w(n)$

$$w(n) = \sum_{k=0}^n a(n-k)u(k) + \sum_{k=0}^{n-1} b(n-k)F[x(k)] \quad (7.2)$$

where  $u(n)$  is the input vector of the original network;

And solves for  $F[x(n)]$  according to equation:

$x(n) = w(n) + b(0)F[x(n)]$  Which can be put in a general form:

$$x(n) = w(n) - BF[x(n)] \quad (7.3)$$

Thanks to the generality of equation (7.3) we have constructed a standard subnetwork N to operate in the equivalent DSP network. The designed subnetwork N provides means to adjust  $w(n)$  and B

depending on the particular application and nonlinear devices. We have shown alternative configurations for one-port  $F[x]$  and two-port  $F[x]$  in first order circuit, second order circuit, and in general,  $n^{\text{th}}$  order circuit. Nonlinear elements with exponential and polynomial characteristics have been specially studied. Its these two nonlinear characteristics which we believe encountered very often in practical applications and hence deserved a detail study. Experiments were built to verify theoretical results for exponential nonlinear elements. The experimental results well agreed with that predicted by theory. With polynomial nonlinear characteristics where the circuit is fairly complex, we used Spice simulation to verify theoretical results.

We have also designed a Programmable Polynomial Module (PPM) which is used as an unit in the standard subnetwork  $N$  to implement an arbitrary nonlinear characteristics. Due to the simplicity of its circuit topology, a PPM can be well fabricated using bipolar microelectronic technology.

A part of this work was devoted to the study of existence and uniqueness of solution for equation (7.3). We have studied in which conditions of original network topology and of nonlinear mapping  $F[.]$  that equation (7.3) possesses solutions and moreover possesses a unique solution. Once a unique solution of equation (7.3) is established then we might construct a digital equivalent network associated with the original analog counterpart with the method proposed in this study. It's in light of work reported by professor Frey in [1] that a general technique for digital implementation from



nonlinear analog networks may ever be established. Such a great technique will certainly make significant contribution to the development of digital signal processing area, and of course involves in more other issues besides that have been addressed in this study. Basically, on the concept of the standard subnetwork N and its related unit to simulate any practical nonlinear characteristics, we believe that a generalized DSP chip can be fabricated for use in nonlinear network application.

## reference

- [1]-Frey, D., private correspondance. CSEE Department, Lehigh University
- [2]-Newcomb, R., "The semistate description of nonlinear time-variable circuits." IEEE circuits and system ,Jan. 81, pp 62-71.
- [3]-Kiyotaka et al, "Solving nonlinear resistive networks by a homotopy method using a rectangular subdivision" IEEE ISCAS'88. pp 1225-28
- [4]-Burkhardt, H., "On nonlinear methods in image processing and pattern recognition " IEEE ISCAS'88, pp 953-55
- [5]-McCannon et al, "On the design of nonlinear discrete-time predictors" IEEE trans. information theory, Mar. 82, pp 366-71
- [6]-Casar-Corredera et al, "Some simple nonlinear prediction schemes" Proceeding of digital signal processing-84, pp 348-52
- [7]-Can, S., "Transfer functions for nonlinear systems via Fourier-Borel transforms" IEEE ISCAS'88, pp 2445-2452
- [8]-Frey, D., "A simple generalized modeling technique for nonlinear networks" IEEE ISCAS'84, pp 343-47
- [9]-Sandberg,I.W. and A. N. Wilson Jr., "Existence and uniqueness of solution of nonlinear DC networks" SIAM j. appl. math., Mar. 72, pp 173-86
- [10]-Frey, D. and O. Norman, "An integral equation approach to the periodic steady-state problem in nonlinear circuits" CSEE Department, Lehigh University

[11]-Gilbert, B., "A precise four-quadrant multiplier with subnanosecond response" IEEE j. solid-state circuits, Dec. 68, pp 365-73

[12]-Duffin, R. J., "Nonlinear network " Bull. Amer. Math. Soc., vol. 53, Oct. 47, pp 10-47

[13]-Desoer, C. A. and J. Katzenelson, "Nonlinear RLC networks" Bell syst. Tech. j., vol. 44, Jan. 65, pp 161-98

[14]-Desoer, C. A. and F. F. Wu, "Nonlinear monotone networks" SIAM j. appl. math., vol. 26, Mar. 74, pp 315-33

[15]-Sandberg, I. W., "Necessary and sufficient conditions for the global invertibility of certain nonlinear operators that arise in the analysis of networks" IEEE trans. circuit theory., vol. 18 ,Mar. 71, pp 260-63

[16]-Wilson, A. N. Jr, "On the solution of equations for nonlinear resistive networks" Bell syst. tech. j., vol. 47 Oct. 68, pp 1755-73

[17]-Sandberg, I. W. and A.N. Wilson Jr., "Some network-theoretic properties of nonlinear DC transistor networks" Bell Syst. tech. J., vol.48, May-June 69 ,pp 1293-311

## **vita**

Trang D. Nguyen was born in 1957. He earned a B.S. degree in Electrical Engineering from Phutho University in Vietnam. He has attended Lehigh University for a M.S. degree in Electrical Engineering since January 1989. His interest field of study is in circuit design and signal processing area.