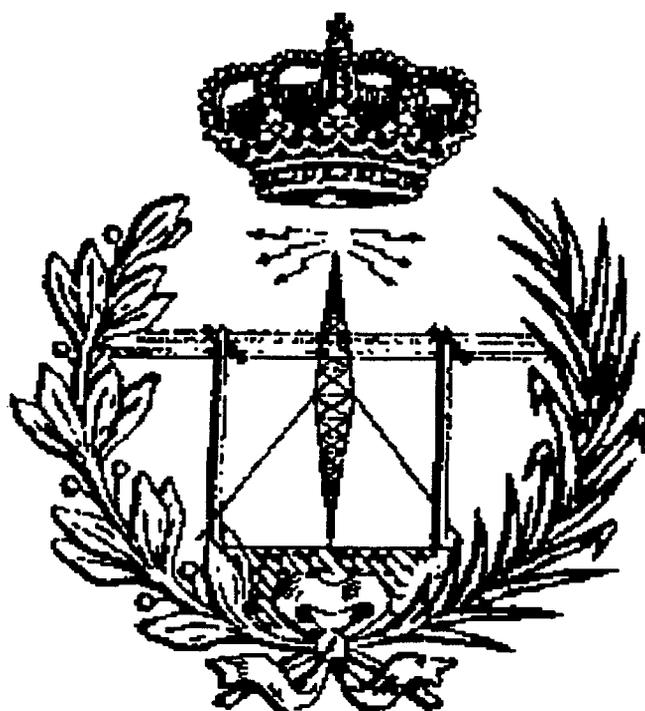


# UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

Escuela Universitaria  
de  
Ingeniería Técnica de Telecomunicación



**TRABAJO FIN DE CARRERA**

**TÍTULO:** Estudio de la Tecnología Bipolar TSFL de Thomson. Aplicación en la realización de una Matriz de Comutación con Control de Ganancia.

**ESPECIALIDAD:** Imagen y Sonido.

**AUTOR:** Francisco José Morera Molina.

**TUTOR:** Roberto Esper-Chaín Falcón.

**COTUTOR:** Juan Manuel Cerezo Sánchez.

**FECHA:** Junio 1995.

# UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

Escuela Universitaria  
de  
Ingeniería Técnica de Telecomunicación



## TRABAJO FIN DE CARRERA

**TÍTULO:** Estudio de la Tecnología Bipolar TSFL de Thomson. Aplicación en la realización de una Matriz de Comutación con Control de Ganancia.

**ESPECIALIDAD:** Imagen y Sonido.

**AUTOR:** Francisco José Morera Molina.

**TUTOR:** Roberto Esper-Chaín Falcón.

**COTUTOR:** Juan Manuel Cerezo Sánchez.

**FECHA:** Julio 1995.

**PRESIDENTE**

**SECRETARIO**

**VOCAL**

**AUTOR**

**NOTA**

**TUTOR**

# Agradecimientos

Durante estos 17 meses son muchas las ayudas desinteresadas, el interés por mi trabajo y los favores que tendría que agradecer. A continuación los enumero aún sabiendo que dejaré alguno olvidado.

- Al laboratorio de VLSI, por el uso de sus dependencias y material, y más concretamente a la gran cantidad de memoria (130 Megas) que he estado utilizando para este trabajo conociendo la escasez de memoria existente en este laboratorio. Especialmente quiero agradecer al encargado del laboratorio, Enrique, por la infinidad de consultas que amablemente me ha atendido.

- A mi tutor, Roberto, por haber puesto a mi disposición todos sus conocimientos, medios, y principalmente su tiempo.

- A Juan Cerezo por su interés y apoyo durante la realización del trabajo.

- A mis padres y mis tías por su constante apoyo.

- A todos los profesores de la EUITT e investigadores del Centro de Microelectrónica Aplicada a los que he abordado para consultarles, prestándome opiniones y soluciones de mucho valor.

- A Lidia, por sus constantes ánimos, y por ayudarme a pasar los malos ratos.

Muchas Gracias a todos.

# Índice

Capítulo 1.- Introducción.....	5
1.1.- Introducción.....	5
1.2.- Antecedentes.....	7
1.3.- Objetivos.....	7
1.4.- Estructura de la Memoria.....	8
Capítulo 2.- Metodología y Tecnología.....	10
2.1.- Metodología.....	10
2.1.1.- Diseño a medida o personalizado.....	10
2.1.2.- Diseño semipersonalizado.....	12
2.1.2.1.- Mar de puertas.....	13
2.1.2.2.- Células estándar.....	15
2.2.- Metodología empleada.....	16
2.2.1.- Los tres arrays de la familia Polyuse L.....	18
2.3.- Tecnología.....	22
2.4.- Tecnología empleada.....	23
Capítulo 3.- Las Herramientas CAD.....	33

3.1.- El Cadence Edge.....	33
3.1.1.- Herramientas de creación de símbolos y esquemas.....	34
3.1.2.- Configuración del sistema para la introducción de circuitos.....	35
3.1.2.1.- Establecer el entorno.....	35
3.1.2.1.1.- Como diseñar el entorno Cadence.....	36
3.1.2.1.2.- Propiedades de la Configuración Global.....	37
3.1.2.1.3.- Propiedades de la configuración de ventanas. ....	38
3.1.2.2.- Establecer el interface de usuario.....	39
3.1.2.2.1.- Menús.....	39
3.1.2.2.2.- Las BindKeys.....	40
3.1.2.2.3.- Entrada de comandos de texto.....	41
3.1.2.3.- Comprensión de los datos del esquema.....	41
3.1.2.4.- Establecer capas.....	43
3.2.- Diseñando en Cadence Edge.....	44
3.2.1.- Configuración de la cuenta de usuario.....	44
3.2.2.- Comienzo de un nuevo diseño.....	46
3.2.3.- Continuar con un diseño existente.....	47
3.2.4.- Captura de esquemas.....	47
3.2.5.- Generación de netlist.....	53
3.2.6.- Simulación.....	55
3.2.6.1.- Simulador Spice.....	56
3.2.7.- Visualizador de formas de onda.....	61
3.2.8.- Realización del layout.....	63

3.2.9.- Herramientas exteriores al Cadence.....	67
3.2.9.1.- Simulador HSPICE.....	68
3.2.9.2.- Visualizador de ondas GSI.....	69
Capítulo 4.- Ruta de Diseño.....	73
4.1.- Simulaciones con Pspice.....	75
4.2.- Simulaciones con Spice.....	78
4.3.- Simulaciones con Hspice.....	87
4.4.- Realización del trazado físico.....	90
4.5.- Comprobación del diseño físico.....	95
4.6.- Elección del encapsulado.....	96
Capítulo 5.- Diseño de células.....	98
5.1.- Buffer de ganancia 1.....	98
5.2.- Amplificador de Banda Ancha.....	107
5.3.- Multiplicador.....	115
5.4.- Multiplexor dos a uno.....	123
5.5.- Operacional.....	134
Capítulo 6.- Diseño de una Matriz de Conmutación con Control de Ganancia.....	140
6.1.- Realización del esquema.....	140
6.2.- Realización del layout.....	145
6.3.- Simulaciones.....	147

Capítulo 7.- Pliego de condiciones.....	153
Capítulo 8.- Conclusiones.....	154
Capítulo 9.- Presupuesto.....	156
Apéndices.....	158
Bibliografía.....	227
Planos.....	230

# Capítulo I

## Introducción

### 1.1.- Introducción.

La integración de circuitos ha dado la posibilidad a la tecnología electrónica de una adaptación necesaria a la evolución de los productos. La tendencia a miniaturizar los productos para su mayor practicidad hace de la microelectronica herramienta necesaria e indispensable en su desarrollo. Gracias a ella en pequeños monocristales de dimensiones del orden de la micra son capaces de contener una cantidad impensable de componentes activos, pasivos y conexiones.

Ventaja notable de la microelectrónica es la excelente identidad en la producción en serie y el bajo coste de éstos. El primer diseño de un circuito integrado puede ser costoso, pero a partir de él se compensará el coste con la producción serie que tiene coste mínimo. Se puede enumerar las siguientes ventajas fundamentales: bajo coste, tamaño reducido, elevada fiabilidad y mejor rendimiento.

En este proyecto se desarrollará diseño microelectronico analógico utilizando tecnología bipolar.

Hasta hace tres décadas la mayoría de los circuitos eran analógicos. El mundo real es básicamente analógico y, en general, un sistema analógico requiere menos componentes que uno digital. Tras la llegada de las microcomputadoras cambió la situación. Mediante técnicas digitales eran realizadas funciones analógicas cada vez mejor. En un principio la desventaja de los sistemas digitales era que requerían una mayor cantidad de componentes y por lo tanto eran mas caros. Los rápidos avances de la tecnología de silicio cambió rapidamente todo esto siendo en la actualidad aproximadamente igual de costoso la construcción de un transistor que de un sistema completo.

Mientras los diseñadores prefieren usar metodos digitales no todos los componentes analógicos pueden ser obtenidos mediante técnicas digitales ya que **"el mundo real es analógico"**. Esta particularidad hace que los interfaces del mundo real al mundo fisico sean analógicos. Así circuitos como los pads de entrada y salida, sensores, amplificadores, osciladores o convertidores deban ser de diseño analógico.

El proyecto que a continuación se presenta pretende mostrar algunas de las posibilidades de el diseño electrónico analógico a traves de la tecnología Polyuse L suministrada por la casa Thomsom a través de EUROCHIP al centro de Microelectrónica Aplicada de la Universidad de Las Palmas de Gran Canaria. Tras una primera introducción a los conceptos generales de microelectrónica pasaremos a demostrar la fiabilidad de las células diseñadas con esta tecnología. Para ello se realizará un estudio de ellas a través de la simulación de éstas. Una vez hecho esto se describirá el diseño del circuito a realizar y se procederá a la simulación, ruteado y verificación para su posterior fabricación.

## **1.2.- Antecedentes.**

El diseño será totalmente bipolar, disponiendo solamente de diferentes tipos de transistores , resistencias , diodos y condensadores. No disponemos de ningun antecedente en este tipo de tecnología siendo el primer diseño con transistores bipolares que se realizará en la Escuela de Ingeniería Técnica de Telecomunicaciones de la ULPGC.

Para una mayor claridad lo desglosaremos segun el tema:

- Tecnología. Es el primer diseño realizado con tecnología bipolar en la Universidad de Las Palmas de Gran Canaria. El único punto de referencia del que se dispone es un pequeño manual suministrado por la casa, el cual no profundiza lo suficiente y solo ha servido , debido a su generalidad como guía de datos técnicos.

- Metodología. No se conoce, en el ámbito de la Universidad, ningun otro diseño realizado exclusivamente con transistores bipolares.

- Herramienta. Las herramientas utilizadas en la ruta de diseño son usadas en la ETSIT y CMA. Con estas se han desarrollado diferentes proyectos y trabajos, obteniendo de ellos, y sus autores, apoyo para el correcto uso de las mismas. El *software* específico de la tecnología es totalmente nuevo, y de él se ha realizado su puesta a punto para poder ser utilizado en diseños posteriores.

## **1.3.- Objetivos.**

El objetivo al que se ha querido llegar con este proyecto es el de sentar unas bases en

la tecnología para posteriores investigaciones o diseños analógicos. El estudio sobre los simuladores analógicos, su uso, su configuración, sus análisis, las técnicas empleadas en el diseño de las células, su modificación según nuestros intereses, hacen que el proyecto sea una herramienta de consulta para posteriores diseños ya que de él se pueden extraer partes perfectamente analizadas para su posterior uso.

### **1.4.- Estructura de la memoria.**

Tras una primera parte donde se muestran las principales características de la tecnología, pasamos a analizar distintas células diseñadas donde presentamos su funcionamiento y parametrización a través de distintas simulaciones. En una tercera parte se describe el proceso de diseño de la Matriz de Conmutación de Alta Frecuencia donde a través de 3 multiplexores enlazados logramos la selección de una señal de entrada a través de una señal de control. Tras acondicionar ésta mediante un buffer para evitar pérdidas al acoplar a la siguiente etapa, hacemos pasar la señal por una etapa de control de ganancia pasando a la amplificación fija final en el amplificador de banda ancha que nos dará la señal de salida.

Este circuito irá encapsulado tras la realización de su *layout* y dispondrá de patillas de entrada, salida, control, verificación de funcionamiento por partes y alimentación. Además ha sido añadido un buffer suplementario independiente al circuito para su posible uso.

La memoria que hemos realizado esta dispuesta en Capítulos distribuidos de la siguiente manera:

El primer capítulo, en el cual nos encontramos, está compuesto por la introducción y antecedentes.

En el segundo capítulo hablaremos de la tecnología y metodología, primero de forma general y posteriormente concretando en la que se va a utilizar en el diseño.

El tercer capítulo está dedicado a la herramienta que hemos utilizado para realizar el diseño, tanto las herramientas dentro del sistema Cadence como las exteriores.

El cuarto capítulo, titulado "Ruta de Diseño" desgloza los pasos a seguir para poder realizar un diseño con la tecnología a la que dedicamos este proyecto.

Los capítulos quinto y sexto constan del diseño de las distintas células que componen el diseño final, y el diseño de la matriz de conmutación al completo.

Al final se incluyen, el Pliego de Condiciones , el Presupuesto , Conclusiones y los Apéndices.

# Capítulo II

## Metodología y Tecnología

### 2.1.- Metodología.

Para el diseño de los sistemas analógicos existen dos metodologías diferentes. Estas son el diseño a medida o personalizado (de la traducción del término inglés *full-custom*) y el diseño semipersonalizado (de la traducción del término inglés *semi-custom*).

#### 2.1.1.- Diseño a medida o personalizado.

En este tipo de metodología el diseñador es el encargado de realizar todo el diseño , desde los componentes a emplear (transistores , resistencias , condensadores , ...) hasta su conexionado.

El elemento básico es el transistor ya sea MOS o bipolar. Un transistor se crea utilizando capas o niveles de diferentes materiales que están disponibles dentro de cada tecnología. Todas esta capas se crean sobre un soporte común que se denomina substrato. Es sobre el substrato donde se difunden impurezas para crear algunas de las capas como son las

difusiones tipo  $n$  o las tipo  $p$ . Además existen otro tipo de capas que se crean sobre el substrato separadas por un óxido, como es el caso del nivel de metal que se utiliza para la interconexión.

Utilizando transistores, resistencias, condensadores y líneas de conexión se consigue plasmar el circuito que previamente se ha diseñado. Se debe comentar que a la hora del diseño, según sea tecnología CMOS, biCMOS ó Bipolar, es ligeramente diferente. Si es CMOS ó biCMOS podemos jugar con las dimensiones de los transistores dependiendo de las necesidades de ancho de banda, cargas capacitivas, niveles de ruido, etc... Si es bipolar nos veremos limitados por el número

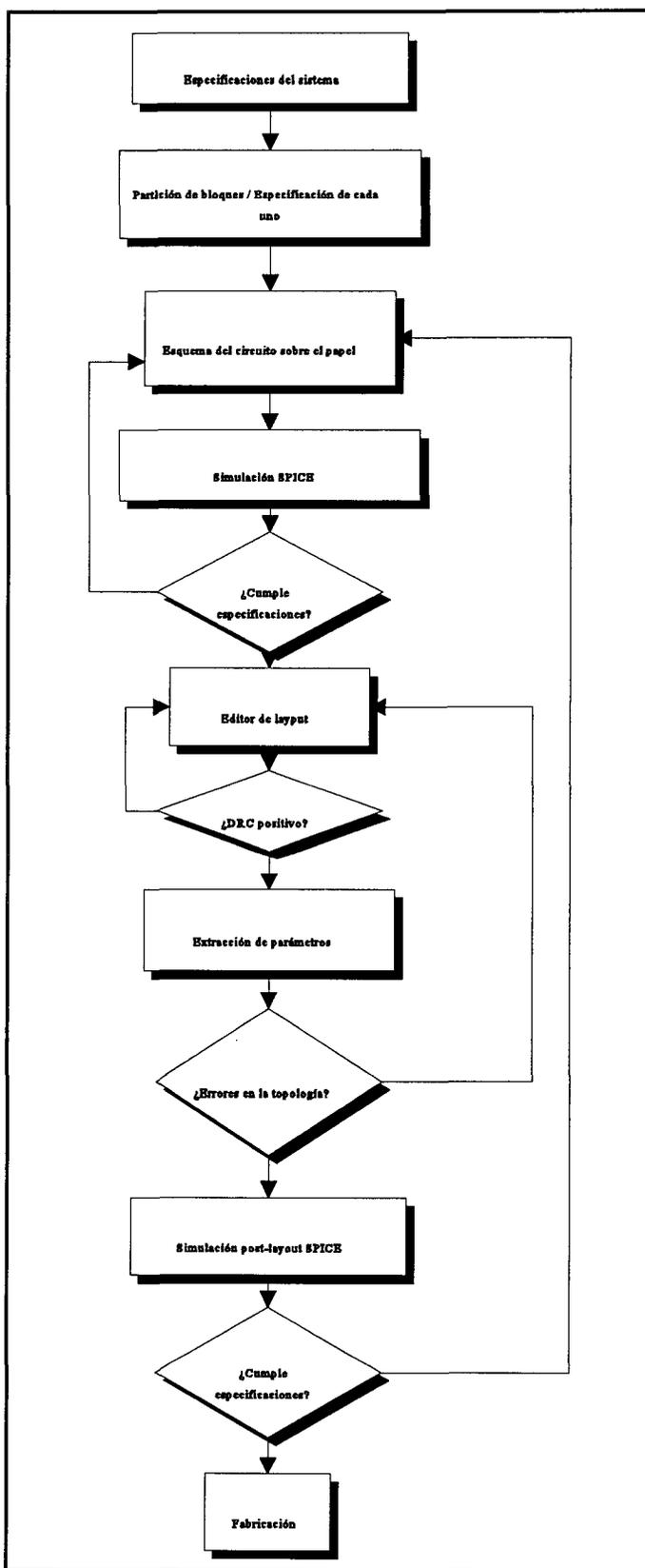


Figura 1

de transistores a utilizar aunque esto no supone una limitación en cuanto a las buenas características que pueden alcanzarse. Por el contrario, cuando se requieren altas prestaciones en ganancia, ancho de banda y ruido, se opta por la solución de un diseño a medida bipolar.

Existen una serie de herramientas *software* que constituyen el entorno de trabajo para un diseño a medida a la vez que definen una secuencia lógica de tareas. En la figura 1 se observa un organigrama donde se resumen los pasos a seguir en un diseño a medida.

Son posibles muchos pasos que no se han descrito aquí, pero los que se han enumerado son comunes a todos los diseños a medida y pueden servir como una guía general.

Todo este proceso de diseño es largo y costoso, por lo que la decisión de realizar un diseño a medida se debe tomar cuando se espera grandes volúmenes de producción. Es solo en este caso cuando se compensan los costes de diseño con la reducción de área que hace bajar el precio del integrado. Con esta reducción de área se consigue, además, una optimización de las prestaciones.

### **2.1.2.- Diseño semipersonalizado.**

Esta metodología es aquella en la que el diseñador realiza una ó mas máscaras de la necesarias para completar todo el proceso de diseño. Dentro de esta metodología vamos a distinguir entre los "mar de puertas" y las "células estándar".

### **2.1.2.1.- Mar de puertas.**

La metodología de diseño mar de puerta también conocida como de redes predifundidas o *gate array* consiste en un circuito integrado que contiene una serie de componentes básicos (transistores, resistencias y condensadores) ó de puertas predifundidas que están dispuestas en el área de diferentes formas. En este caso se tendrá un número limitado y conocido de componentes y el trabajo del diseñador consistirá en determinar las interconexiones entre ellos con un nivel de metal.

Si el principal componente es el transistor reciben el nombre de mar de transistores (*transistor arrays*). Esta metodología se ha restringido clásicamente a tecnologías bipolares. Solo se disponen de transistores *npn* optimizados, mientras los transistores *pnp* presentan menor ganancia de corriente y menor frecuencia de corte. Por este motivo son los transistores *npn* los más utilizados aunque ello suponga un incremento del número de transistores del circuito. Por este motivo, es habitual que el número de transistores *npn* que contiene el dado sea mayor que el de *pnp*.

Las resistencias se obtienen normalmente con una capa de difusión *p*. Puede existir un gran número de resistencias para facilitar la combinación entre ellas, haciendo el diseño más flexible.

Los condensadores se pueden realizar mediante las uniones *pn* inversamente polarizadas de los transistores.

Como podemos observar, existen frente a la metodología de diseño completo ó a medida fuertes limitaciones en cuanto al número de dispositivos disponibles. Esto hace que la utilización de esta metodología sea para la realización de subsistemas y no de sistemas completos.

La filosofía de diseño de un mar de puerta consiste en que el fabricante proporcione los bloques elementales con los que construir un diseño, utilizando si es necesario el resto de componentes como transistores, resistencias, condensadores,... En un principio no se caracterizaban por la repetitividad de los componentes por lo que se crearon las células analógicas conocidas como "tiles". Estas son celdas de igual tamaño y contenido de componentes que se repiten uniformemente a lo largo de todo el

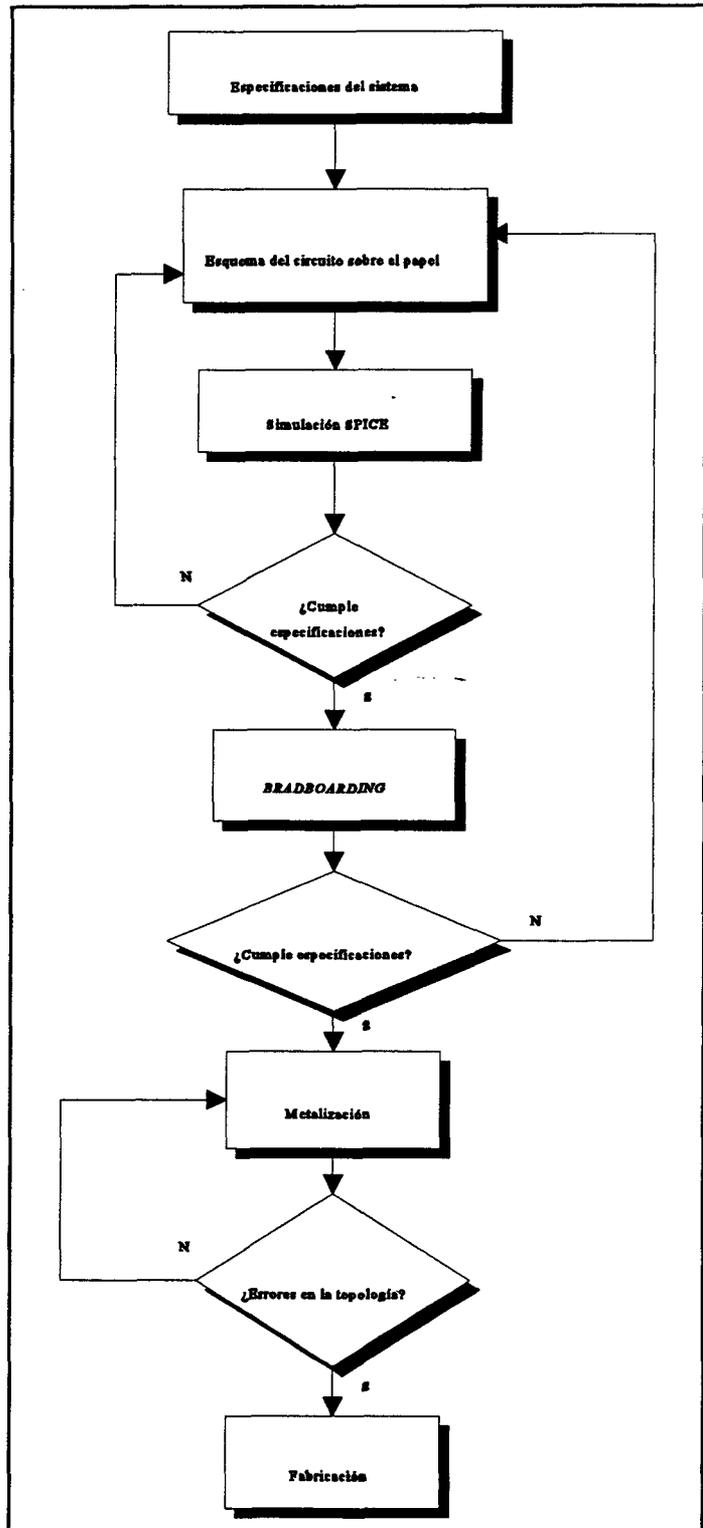


Figura 2

dado. Sus componentes internos pueden ser muy diversos, pero en definitiva se trataba de una serie de transistores *npn* y *pnp*, resistencias de diferentes tipos e incluso algun condensador. De esta forma se lograba facilitar el uso de macrocélulas al lograr la uniformidad en el dado. Cada macrocélula podrá implantarse con un número determinado de *tiles* dependiendo de su complejidad. Con este sistema se consigue un ahorro de superficie de silicio , puesto que el diseñador determina el número de *tiles* que necesita y solo éstos son los que se incluyen el integrado. Esto no evitará que casi siempre se pierda alguno de los componentes dispuestos en la superficie. El último paso en este tipo de metodología consistió en que los fabricantes ponen a disposición del diseñador una serie de *tiles* que se diferencian en su contenido. El diseñador elige según su conveniencia teniendo la posibilidad de modificarlos si lo estima conveniente para su diseño. Este tipo de diseño se le conoce como estandar *tiles*.

El proceso de diseño de un mar de puerta sigue en general los pasos desglosados en el organigrama de la figura 2.

Este proceso de diseño suele ser corto, por lo que los mar de puertas suelen ser utilizados para pequeños volúmenes de producción donde el precio del circuito integrado no es alto. También resulta muy conveniente la utilización de esta tecnología cuando se necesita que el circuito integrado tenga una rápida aparición en el mercado.

#### **2.1.2.2.-Células estándar.**

En el diseño con células estándar no existen elementos predifundidos en el integrado. El fabricante pone a disposición del diseñador una librería de células previamente definidas y caracterizadas en sus parámetros eléctricos y físicos de las cuales el diseñador elegirá y situará en el integrado según convenga.

Al tener el diseñador la posibilidad de elección y colocación de las células a emplear el aprovechamiento del área es mayor que en un mar de puerta y por lo tanto el costo de producción será menor si bien al tener que diseñar mayor número de capas implica un mayor costo de desarrollo.

Los tiempos de realización de prototipos son mayores que los de un mar de puerta lo que lógicamente supone una desventaja para quien desee rapidez en el diseño.

La mayor ventaja sobre los mar de puerta es la flexibilidad del diseño ya que por lo general el número de células diseñadas por el fabricante suele ser mucho mayor de las que disponemos en un mar de puerta.

## **2.2.- Metodología empleada en este trabajo.**

La POLYUSE L es una familia de *arrays* mixtos analógicos/digitales disponibles en rangos de temperatura tanto militar como comercial y están fabricados según las normas estándar MIL883 , CECC y ESA.

Se dispone de 3 *arrays* , L06, L12, y L19, contruidos y uniformemente distribuidos en grupos de componentes (Q, R, C) llamados *tiles* los cuales permiten una fácil realización del *layout* y flexibilidad de colocación.

La primera capa de metal esta predefinida (distribución de la potencia de alimentación, cruces y polarización) y solo la segunda capa es la que el diseñador realizará obteniendo las siguientes ventajas:

.-Reducción del coste y ciclo de tiempo.

.-Flexibilidad en el *layout* de la estructura de doble capa de metal.

Observaremos en el capítulo dedicado a la realización del *layout* como esta ventaja es relativa , ya que la realización del *layout* no la consideraremos totalmente flexible.

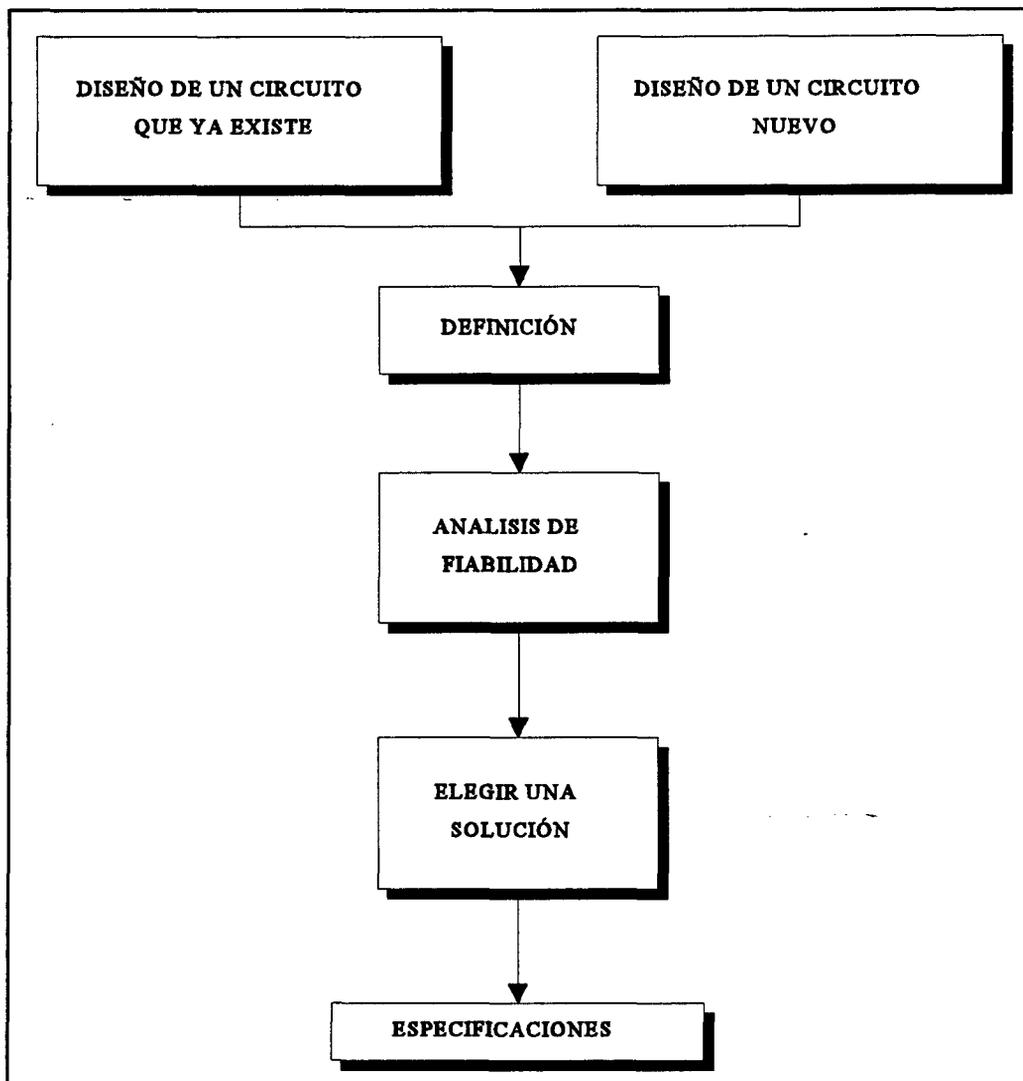
.-Conocimientos de silicio mínimos para poder realizar el diseño.

.-Efectos parásitos son minimizados debido al diseño con la segunda capa de metal.

.-Funciones macros disponibles en una librería. (Esta ventaja no la hemos podido utilizar al no disponer el *software* donde se incluyen).

Los *arrays* han sido desarrollados para facilitar la tarea del diseño de proyectos y mas especialmente, para simplificar el esquema electrico del *layout*.

Los fabricantes de esta tecnología nos dan la posibilidad de la implementación de nuestro diseño. Para ello nosotros debemos enviarles nuestro diseño con las simulaciones correpondientes, el *layout* y el ruteado de nuestro diseño.



**Figura 3**

El diagrama de bloques de la figura 3 ilustra los ciclos de la evolución del diseño típico de un proyecto.

### **2.2.1.-Los tres arrays de la familia Polyuse L.**

Los tres tipos diferentes de *arrays*, L06, L12 y L19, se diferencian esencialmente por su tamaño, la superficie, el número de componentes y el número de patillas.

Todos los tipos comparten características comunes tales como arquitectura celular y componentes básicos de construcción.

Los *arrays* de Polyuse L están divididos en casillas individuales repetidas a lo largo de toda el área las cuales permiten:

- Fácil colocación de las macrocélulas disponibles dentro de la librería (no suministradas en el *software*).
- Fácil copia de una función para llevar a cabo múltiples funciones.
- Rápido y fácil aprendizaje del contenido de las casillas para el diseño. Tras su utilización varias veces, el diseño será mas rápido y cómodo debido a la mejor planificación que se realizará.

La siguiente tabla nos muestra el contenido de los diferentes tipos de casillas, internas, externas y centrales en cada uno de los tipos de *arrays* de los que disponemos actualmente en Polyuse L.

		L06	L12	L19
<b>NUMERO DE CASILLAS ( TILES )</b>		16	30	48
<b>CASILLAS</b>	QN1 (NPN 1 ma) optimo	80	150	240
	QN2 (NPN 2 ma) optimo	32	60	96
	QPL (lateral PNP 50 uA)	48	90	114
<b>INTERNAS</b>	Resistor (P+)	224	420	672
	Resistor (P base extrínseca)	256	480	768
<b>CASILLAS</b>	Condensadores 3 pF	NA	NA	12
<b>CENTRALES</b>	DS1 Diodo Schottky (500 uA)	NA	NA	24
	QPL 12 Lateral PNP (500 uA)	NA	NA	6
<b>CASILLAS</b>	Condensadores 3 pF	12	8	8
	Condensadores 6 pF	4	8	12
	QNS 20 NPN de potencia 200 mA	NA	NA	8
<b>PERIFERICA</b>	QN4 (NPN 40 mA) max.	16	16	8
	Resistor de 25 ohm. (P+)	8	8	16
	QPS PNP substrato (10 mA)	4	4	4
	QPS 025 PNP subst. (2,5 mA)	NA	NA	8
	Cantidad de componentes	640	1180	2026
	Vías	2	3	3
	Patillas	20	28	40
	Resistencia total	1.5M	2.9M	4.6M

Cada casilla de Polyuse L consta de un número uniforme de componentes. Los bloques de una casilla básica están listados a continuación.

<b>DESCRIPCIÓN</b>	<b>CANTIDAD</b>
<b>"QN1"</b> Transistor NPN	5
<b>"QN2"</b> Transistor NPN	2
<b>"QPL"</b> Transistor PNP	3
<b>100 ohm.</b> Resistencia ( P+ )	2
<b>200 ohm.</b> Resistencia ( P+ )	4
<b>400 ohm.</b> Resistencia ( P+ )	2
<b>800 ohm.</b> Resistencia ( P+ )	6
<b>1K</b> Resistencia ( P base extrínseca )	2
<b>2K</b> Resistencia ( P base extrínseca )	4
<b>4K</b> Resistencia ( P base extrínseca )	4
<b>8K</b> Resistencia ( P base extrínseca )	4
<b>16K</b> Resistencia ( P base extrínseca )	2
<b>Canal de Ruteado</b>	13
<b>Conexiones de Alimentación Negativa</b>	2
<b>Conexiones de Alimentación Positiva</b>	4

Los transistores de alta corriente *PNP* y *NPN*, resistores de bajo valor y todos los condensadores no están situados dentro de las células sino en la periferia de los *arrays* entre el patillaje. En el array L19 estan localizados dentro de una sección especial en la mitad del *array* igual de válida que en la periferia.

### **2.3.- Tecnología.**

-Las tecnología mayormente utilizadas hoy en día en el diseño semipersonalizado ó *semicustom* son la CMOS y la Bipolar.

La tecnología bipolar proporciona, en general, más altas prestaciones aunque un mayor consumo. El uso de CMOS es especialmente apropiado en sistemas de datos muestreados y, en general, en todos aquellos sistemas en que exista una fuerte interconexión entre funciones digitales y analógicas.

Caracterizando un poco a estas tecnologías podemos decir que en tecnología CMOS podemos conseguir condensadores con una precisión del 0,1% y resistencias con precisión del 2% mientras que para tecnología bipolar las capacidades no alcanzan tanta precisión como para CMOS y las resistencias pueden alcanzar precisiones tan bajas como el 0,2%, aunque para valores de resistencia bajos.

Problema crítico en CMOS es que un transistor típico no puede proporcionar mas que unas decenas de miliamperios a una tensión de +/- 5 voltios.

Para una tecnología CMOS podemos jugar con las dimensiones de los transistores dependiendo de las necesidades, sin embargo, para una tecnología Bipolar existe un número limitado de transistores con los que se está obligado a realizar el circuito. A pesar de esto cuando se requieren altas prestaciones en ganancia , ancho de banda y ruido , la solución óptima pasa por una tecnología bipolar.

Actualmente están surgiendo tecnologías BiCMOS que combinan lo mejor de las características de bipolar y CMOS. Esta tecnología permite la utilización de transistores bipolares donde se necesite velocidad de funcionamiento ya sea analógico ó digital , mientras donde se necesite alta densidad de integración y bajo consumo se utilizarán transistores CMOS.

## **2.4.- Tecnología empleada.**

La tecnología empleada en el diseño es el proceso de silicio HF2C que consiste en una tecnología bipolar la cual suministra dispositivos de alto rendimiento para aplicaciones de alta frecuencia y bajo nivel de ruido. Como ejemplo diremos que las características del transistor *NPN* son las siguientes:

FRECUENCIA DE CORTE: 3 Ghz.

BETA: 100 ( $I_c=1$  mA.)

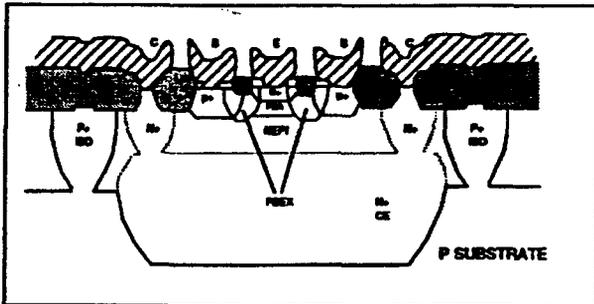
NIVEL DE RUIDO: 4nV/Hz.

MÁXIMO VOLTAJE DE ALIMENTACIÓN: 15 V.

La tecnología nos pone a disposición los modelos Spice de los componentes disponibles dentro de los *arrays* Polyuse así que podremos simular el diseño usando las herramientas que elijas: IBM PC , VAX , WORKSTATION ,... SPICE es un *software* de simulación de circuitos electrónicos que será estudiado con mayor profundidad mas adelante.

A continuación vamos a desglosar las características y estructuras de los componentes que están dispuestos en los *arrays*. La leyenda será la siguiente:

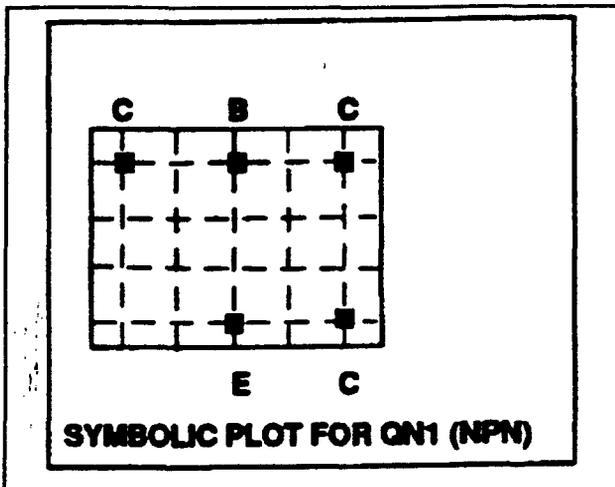
- E , B , C: Contactos de emisor , colector y base.
- ISO: Difusión de aislamiento.
- CE: Burried layer
- PBA: Difusión de base tipo *p*.
- PBEX: Difusión extrínseca de base tipo *p*.
- NEPI: Capa epitaxial *n*.
- N + EM: Difusión de emisor tipo  $N^+$



Estructura de capas de QN1.

.- Transistor tipo NPN ,  
"QN1".

El QN1 es un transistor de pequeña señal con una zona de emisor y dos contactos de base.



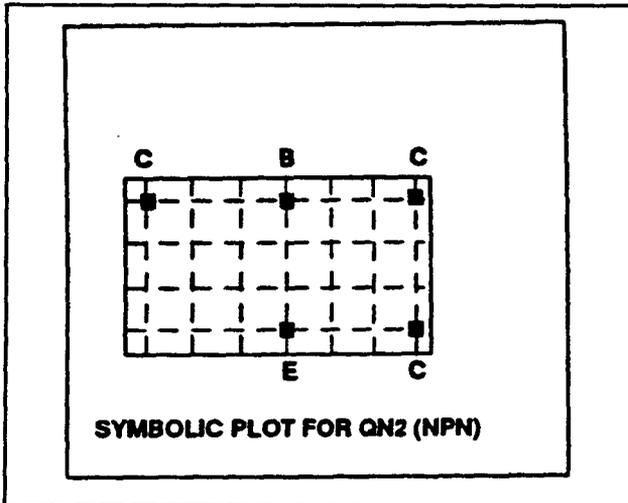
Símbolo del transistor QN1.

La resistencia de acceso a base "rbb" es aproximadamente de 250 Ohmios.

- Voltaje de Ruptura colector-base:  $V(BR)_{CBO}=25$  Voltios.

- Voltaje de Ruptura colector emisor:  $V(BR)_{CEO}=15$  Voltios.

- Voltaje de ruptura colector sustrato:  $V(BR)_{CSO}=25$  Voltios.
- Voltaje de ruptura emisor base:  $V(BR)_{EBO}=5,8$  Voltios.
- La máxima corriente de colector es 10 mA.

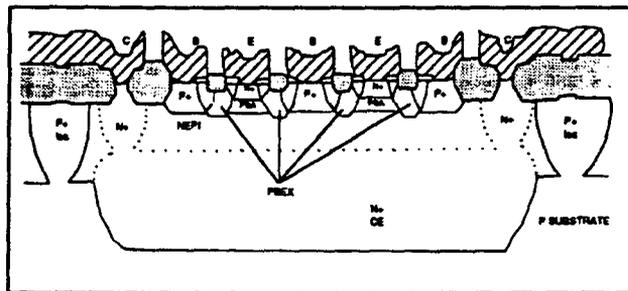


Simbolo del transistor QN2.

.-Transistor NPN,

"QN2".

El transistor "QN2" tiene dos zonas de emisor y tres contactos de base. La resistencia de acceso a base "rbb" es de aproximadamente 200 Ohmios. El area superficial de emisor es



Sección transversal de QN2.

igual que la de "QN1".

La máxima corriente de colector es 10 mA.

Voltaje de ruptura colector-base:  $V(BR)_{CBO}= 25$  Voltios.

Voltaje de ruptura colector emisor:  $V(BR)_{CEO}= 15$  Voltios.

Voltaje de ruptura colector substrato:  $V(BR)_{CSO} = 25$  Voltios.

Voltaje de ruptura emiso base:  $V(BR)_{EBO} = 5,8$  Voltios.

**.- Transistor tipo NPN , "QN4".**

El "QN4" es un transistor de bajo ruido con cinco zonas de emisor y seis contactos de base.

Resistencia de acceso a base "rbb" aproximadamente 100 Ohmios.

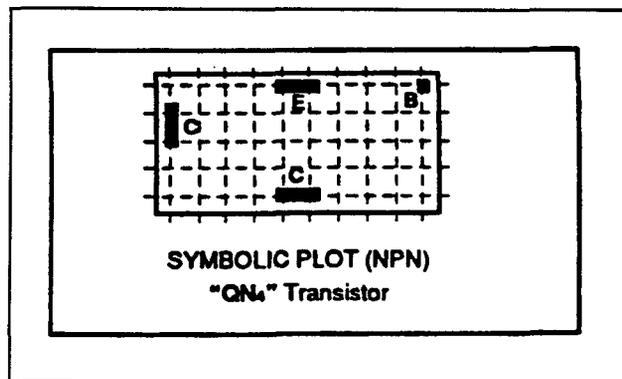
El área de la superficie de emisor es 5 veces mas grande que la de "QN1".

La máxima corriente de colector es de 40 mA.

Voltaje de ruptura colector-base:  $V(BR)_{CBO} = 25$  Voltios.

Voltaje de ruptura colector emisor:  $V(BR)_{CEO} = 15$  Voltios.

Voltaje de rupturacolector

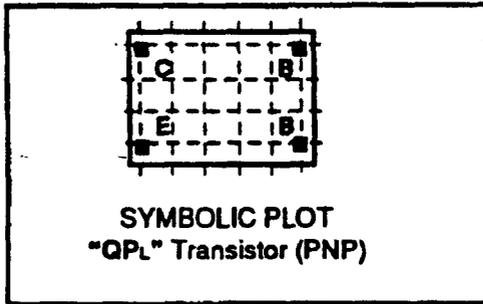


Símbolo del transistor QN4.

substrato:  $V(BR)_{CSO} = 25$  Voltios.

Voltaje de ruptura emiso base:  $V(BR)_{EBO} = 5,8$  Voltios.

El diagrama de la seccion nos muestra como la estructura de capas es igual solo que tiene 5 zonas de emisor y 6 contactos de base.

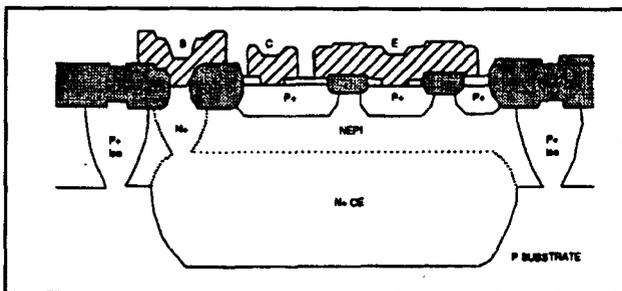


Símbolo del transistor QPL.

- Transistor tipo PNP ,  
"QPL".

Los transistores PNP solo son eficientes para baja frecuencia ( $f_T=10$  MHz.). Se usaran para etapas de funcionamiento en continua

(fuentes de corriente) ó en modo dinámico pero para bajas frecuencias.



Sección transversal de QPL.

El "QPL" es un transistor lateral PNP. Su área de emisor está completamente rodeada por su zona de colector.

La máxima corriente de colector es alrededor de 0,1 mA.

Voltaje de ruptura colector-base:  $V(BR)_{CBO}=25$  Voltios.

Voltaje de ruptura colector emisor:  $V(BR)_{CEO}=15$  Voltios.

Voltaje de ruptura colector substrato:  $V(BR)_{CSO}=25$  Voltios.

Voltaje de ruptura emiso base:  $V(BR)_{EBO}=25$  Voltios.

**.-Transistor tipo PNP, Qps.**

Voltaje de ruptura colector-base:  $V(BR)_{CBO}=25$  Voltios.

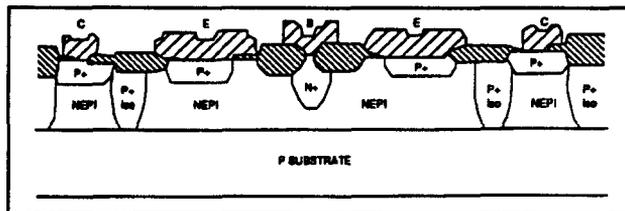
Voltaje de ruptura colector emisor:  $V(BR)_{CEO}= 15$  Voltios.

Voltaje de ruptura colector substrato:  $V(BR)_{CSO}= 25$  Voltios.

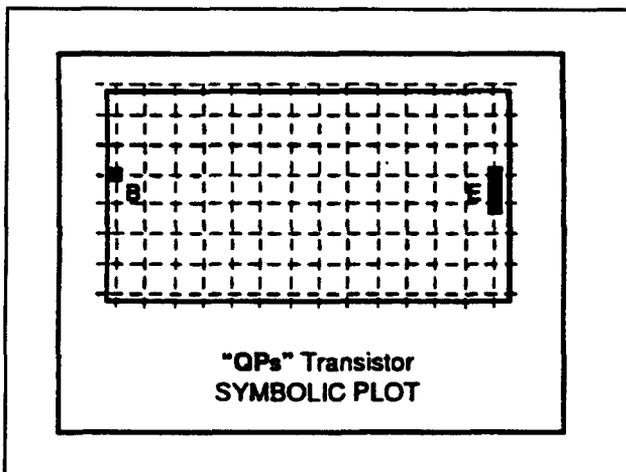
Voltaje de ruptura emiso base:  $V(BR)_{EBO}= 25$  Voltios.

El "QPs" es un transistor PNP de sustrato que puede ser usado en etapas de salida.

Su región de base es similar a la región de colector de un transistor NPN. Su



Sección transversal QPs



Símbolo del transistor Qps.

colector está conectado al sustrato. Su frecuencia de corte esta en el rango de los 50 MHz así que lo usaremos para rangos de frecuencia moderados.

**.-Resistor tipo p<sup>+</sup>.**

Sus valores varían desde 100 a 800 ohmios.

Tolerancia +/- 25%.

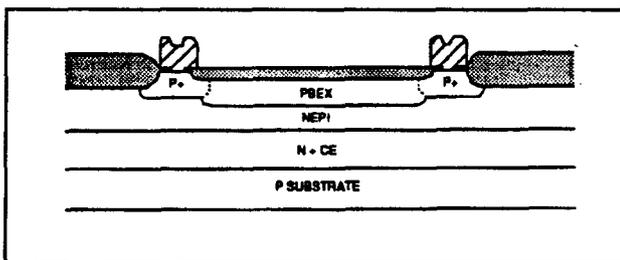
Coefficiente de temperatura dado por:

$$R=R_0(1+TC1(T-T_0)+TC2(T-T_0)^2) \text{ donde,}$$

$$TC1= 1,25*10^{-3}$$

$$TC2= 9,47*10^{-7}$$

$R_0$  = Resistencia a la Temperatura inicial.



Sección transversal del Resistor p<sup>+</sup>.

La capacidad parásita de los resistores viene dada por la expresión:

$$C=C_0(1+V/O)^{-m} \text{ donde,}$$

V es el voltaje entre el resistor y masa.

$$O= 0,65.$$

$$m= 0,3.$$

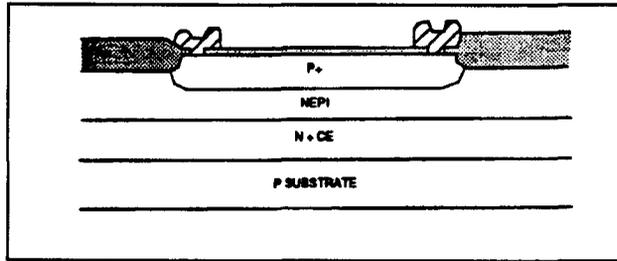
$$C_0= 50 \text{ fF} + 90 \text{ fF/Kohmio} +R.$$

Los resistores p<sup>+</sup> se implatan a la vez que las zonas p de los transistores. Los valores disponibles de resistencias de este tipo son : 100, 200, 400 y 800 Ohmios.

#### **.-Resistor tipo RBEX.**

Sus valores variarán desde 1 Kohmio hasta 16.

Su tolerancia es del 25%.



Sección transversal del resistor RBEX

El coeficiente de temperatura viene dado por la misma expresión que el del tipo anterior, aunque varían los valores de las constantes.

$$TC1 = 5,13 \cdot 10^{-4}$$

$$TC2 = 6,8 \cdot 10^{-6}$$

La capacidad parásita también viene dada por la misma expresión y los valores de las constantes son los siguientes:

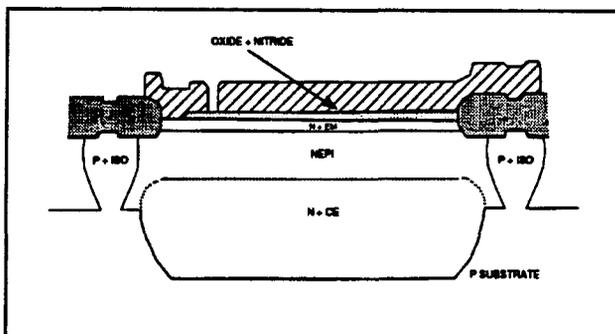
$$O = 0,7.$$

$$m = 0,45.$$

$$C_0 = 80 \text{ fF} + 14 \text{ fF/Kohmio} \cdot R.$$

Los resistores PBEX son difundidos al mismo tiempo que la base de los transistores NPN. Los valores posibles son de 1, 2, 4,8 y 16 Kohmios.

**.- Condensadores.**



Sección transversal del condensador.

Hay disponible un tipo simple de condensador. Es un condensador MOS implementado por una capa de aluminio.

Los valores van a variar des los 3 a los 6 pF (20%).

Maximo voltaje de trabajo= 20 Voltios.

A continuación se muestran una serie de tablas indicando las características de los transistores, resistencias y codensadores de las que hemos hablado anteriormente.

### Características de los Transistores

TRANSISTOR	Rbb	Icmax	Vcbo	Vceo	Vcso	Vebo	beta
QN1	250	10 mA	25 V.	15 V.	25 V.	5,8 V.	125
QN2	200	10 mA	25 V.	15 V.	25 V.	5,8 V.	125
QN4	100	40 mA	25 V.	15 V.	25 V.	5,8 V.	125
QPI	--	0.1 mA	25 V.	15 V.	25 V.	25 V.	50
QPs	--	30 mA	25 V.	15 V.	25 V.	25 V.	315

**Características de las Resistencias**

RESISTENCIAS	P <sub>MAX.</sub>	I <sub>MAX.</sub>	V <sub>MAX.</sub>
1 K	12 mW	3.4 mA	3.4 V
2K	24 mW	3.4 mA	6.8 V
4K	48 mW	3.4 mA	13.6 V
8K	30 mW	2 mA	15 V
16K	16 mW	1 mA	15 V
25 Ohmios	40 mW	40 mA	1 V
100 Ohmios	10 mW	10 mA	1 V
200 Ohmios	18 mW	10 mA	2 V
400 Ohmios	36 mW	10 mA	4 V
800 Ohmios	72 mW	10 mA	8 V

# Capítulo III

## Las herramientas CAD

Todo diseño de circuito integrado debe tener un *software* de soporte que será el encargado de darnos la posibilidad de diseñar y simular nuestro circuito. Si no dispusiéramos de este *software*, nos quedaría la posibilidad de encargar el diseño a alguna de las empresas existentes dedicadas al diseño y fabricación de circuitos integrados. En este caso deberíamos darles las especificaciones y ellos se encargarán de realizar todos los pasos necesarios hasta el diseño fabricado.

El gran inconveniente es que esta opción resultará excesivamente costosa por lo que debemos tratar de realizar la mayor parte de nuestro diseño con métodos propios y así dejar solo lo imprescindible para la empresa dedicada.

La herramienta que vamos a utilizar es el **CADENCE/Edge** instalada en el Laboratorio de VLSI de la Universidad de Las Palmas de Gran Canaria.

### 3.1- El Cadence/Edge.

La utilización de Cadence/Edge como entorno de diseño base permitirá realizar

diferentes tareas de diseño, desde la captura de esquemas y simulación hasta el trazado físico y su verificación.

A lo largo de un proceso de diseño, Cadence suministra una serie de módulos para la creación y edición de esquemas, símbolos y librerías. Además suministra documentación gráfica de interés para el diseñador, entradas de esquemas electrónicos, base de datos del diseño usadas en la simulación, el diseño físico y la verificación de herramientas del sistema. El beneficio más destacable del Cadence es la integración de todas las herramientas del sistema.

Podremos usar una serie de herramientas que simplifiquen el proceso de diseño e incrementen su productividad cuando realicemos el diseño usando el sistema Cadence.

Usando el entorno de multi-ventanas se puede editar múltiples esquemas, símbolos y *layouts*. Los esquemas Cadence ofrecen la posibilidad de ser fácilmente caratulizados y además usando la jerarquía esquemática y el uso de buses reducen significativamente la complejidad de éstos.

### **3.1.1.-Herramientas de creación de símbolos y esquemas.**

Además de las herramientas de edición gráfica, Cadence dispone de herramientas para la creación automática de símbolos y librerías.

\* El generador de esquema-símbolo de Cadence (SSG) crea símbolos para

esquemas existentes.

\* El generador de texto-símbolo de Cadence (TSG) crea símbolos desde texto formateado de tal forma que se podrá transferir automáticamente símbolos de librería desde otros sistemas dentro de Cadence.

\* El generador de librerías símbolo-simulación (S/SLG) dispone la abilitación y creación de un listado de todos los bloques de una librería, incluyendo sus pines y propiedades.

Podemos usar los comandos de edición gráfica para modificarlo o diseñar los símbolos creados por SSG y TSG. Podemos usar las herramientas de edición de texto para modificar o añadir propiedades en la librería.

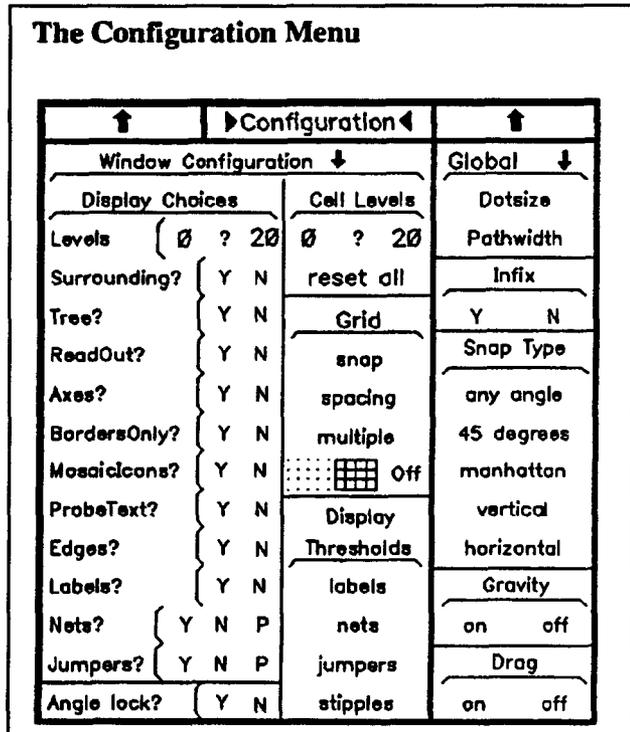
### **3.1.2.- Configuración del sistema para la introducción de circuitos.**

Antes de comenzar con la creación o edición de un esquema, necesitamos alguna información sobre el sistema Cadence:

- Establecer el entorno.
- Establecer el interface de usuario.
- Comprensión de los datos del esquema.
- Establecer capas.

#### **3.1.2.1.- Establecer el entorno.**

Las propiedades que controlan la interacción del sistema con el usuario definen la configuración del sistema. Hay propiedades que se aplican a todas las ventanas gráficas del sistema y otras propiedades que se aplican a ventanas específicas. La primera se llama Configuración Global y la segunda Configuración de Ventana.



El fichero .setenv situado en el directorio cds contiene la configuración por defecto tanto global como de ventana. **Figura 3.1 Menú de Configuración.**

en el directorio cds contiene la configuración por defecto tanto global como de ventana.

El fichero .sdarc situado en el directorio del usuario contiene la configuración global, las bindKeys (macros de teclado),...constituyendo lo que se conoce con el nombre de fichero de configuración local.

### 3.1.2.1.1.-Cómo diseñar el entorno Cadence.

Lo primero que necesitamos diseñar es la configuración global y de ventana. Además de las propiedades por defecto especificadas en .setenv podemos añadir en el fichero .sdarc las bindKeys y ejecutar otros comandos Cadence. Otra posibilidad es la especificación de los directorios donde están localizados las librerías modificando el valor de

la propiedad *blockpath* correspondiente a la configuración global.

Las variables de entorno de ventana pueden modificarse sobre el fichero de diseño.

Una lista de las propiedades de configuración está disponible en el menú de configuración que mostramos en la figura 3.1. En este menú debemos colocar el cursor en Propiedades de Configuración Global ó Propiedades de Configuración de Ventana según deseemos y ejecutarlo.

Si el cambio en la configuración es permanente es recomendable incluirlo en el fichero *.sdarc*. Cuando invoquemos el sistema Cadence desde algun directorio el sistema usará el *.sdarc*.

#### **3.1.2.1.2.-Propiedades de la Configuración Global.**

Si seleccionamos el comando Global del menú de configuración podremos ver la totalidad de las propiedades de la Configuración Global en una ventana de edición de lista de propiedades. La ventana de propiedades suministra un interface para la edición de propiedades.

La siguiente tabla muestra los valores de Configuración Global recomendados para la captura de esquemas.

<b>Propiedad</b>	<b>Valor</b>
selectWhenprobed?	No
gravityOn?	Si
representationPath	schematic
font	stick
fontHeight	0.125
dotSize	0.0625
pathWidht	0.0625
labelLayer	wire
masterRepresentationName	symbol
representationName	schematic
snap	manhattan

**3.1.2.1.3.-Propiedades de Configuración de Ventana.**

Si seleccionamos el comando de Configuración de Ventana obtendremos las propiedades de la ventana en donde estemos situados, de forma similar a las globales.

La siguiente tabla muestra los valores de Configuración de Ventana recomendados para la captura de esquemas. Éstos valores son los que se dan por defecto al arrancar el sistema. Como se puede observar los valores dados se podran variar segun sea el aspecto que queramos dar a la ventana de visualización,o los elementos que deseemos visualizar.

<b>Propiedad</b>	<b>Valor</b>
snapSpacing	0,0625
screenGridSpacing	0,125
screenGridMultiple	8
displayAxes?	Yes
displayGrid?	Yes
displayLabels	Yes
displayLabelsWhen	0,125
displayJumpers?	No
displayNets?	No
displayStartLevel	0
display StopLevel	20

### **3.1.2.2.- Establecer el interface de usuario.**

Basado en el sistema de comandos antiguos, una completa serie de menús y *bindKeys* constituyen el interface de usuario en las herramientas de Cadence.

Tenemos la posibilidad de modificar, reemplazar, o añadir a los menús y *bindKeys* existentes. Una vez hayas logrado familiarizarte con las herramientas y el interface de usuario puedes elegir y modificar el interface de usuario según tus preferencias y estilo de diseño.

#### **3.1.2.2.1.- Menús.**

Son el camino mas fácil para introducir comandos. Para introducir un comando desde un menú realizaremos lo siguiente:

- 1.- Presionar el botón izquierdo del ratón para ver el menú.
- 2.- Posicionar el cursor sobre el comando a ejecutar y presionar el botón izquierdo o central del botón.

Los menús de Cadence están estructurados de tal forma que puedes llamar a otros menús desde el que tienes en pantalla. La figura 3.2 muestra el árbol completo usado para edición de esquemas y símbolos.

Cuando inicialmente llamamos al sistema Cadence el primer menú que podemos observar es el "Design Manager". Desde éste puedes moverte por el árbol entero. Selecciona la flecha hacia abajo para moverte hacia abajo en la jerarquía de menús y la flecha hacia arriba para moverte ascendentemente en la jerarquía.

Los menús pueden estar cargados en varios directorios. La variable *menuPath* de la Configuración Global debe especificar estos directorios. Puedes cargar menús propios incluyendo el directorio en el *path*.

#### **3.1.2.2.2.- Las BindKeys.**

Las bindKeys son macros de teclado que ejecutan un comando con solo la

pulsación de esta tecla.

Las *bindKeys* están definidas en el fichero *.sdarc* y estas pueden ser modificadas a nuestro criterio. Como es lógico la utilidad de las *bindKeys* adquiere una mayor importancia a la hora de ejecutar comandos muy comunes.

### **3.1.2.2.3.- Entrada de comandos de texto.**

Podemos ejecutar comandos directamente en el sistema Cadence. Para ello debemos ejecutar la combinación de teclas CTRL V obteniendo con ella una pantalla de entrada de texto, la cual utilizaremos para la introducción de comandos por el teclado.

### **3.1.2.3.- Comprensión de los datos del esquema.**

Las herramientas del sistema Cadence utilizan la misma base de datos. La unificación de conceptos en la base de datos del Cadence se llama Bloque (Block).

El bloque describe funciones electrónicas que pueden ser desde un simple transistor o puerta lógica hasta un complejo microprocesador o sistemas de comunicaciones.

Los bloques pueden tener un número de representaciones diferentes, las cuales describen un tipo particular de datos del bloque.

Por ejemplo, la representación *layout* describe datos de la composición de máscaras sobre el bloque, la representación esquema describe la conectividad y la representación símbolo describe solo los puertos de entrada/salida del bloque y alguna información global sobre éste.

La base de datos del Cadence puede mantener diferentes revisiones de cada representación del bloque, usando siempre por defecto la revisión "*current*".

La base de datos del Cadence usa el sistema de administración de ficheros del UNIX para almacenar los bloques, representaciones y revisiones en forma de directorios, subdirectorios y ficheros, respectivamente. Un bloque es un directorio UNIX con el mismo nombre que el bloque; una representación es un subdirectorio bajo el directorio del bloque.

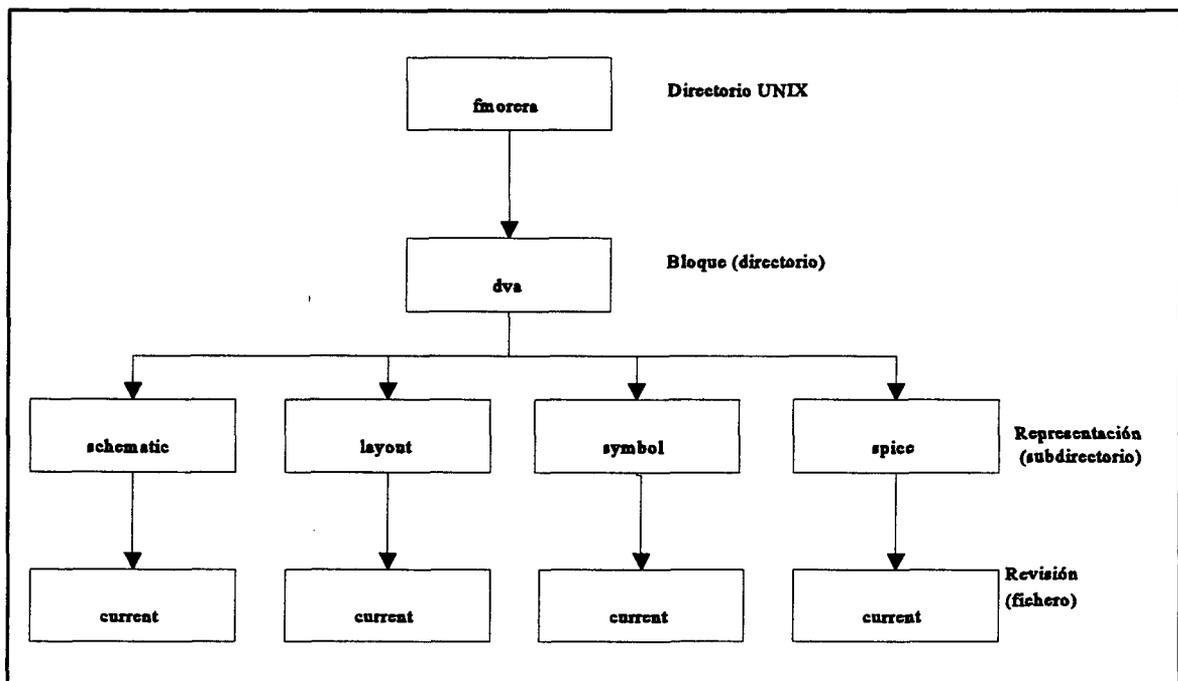


Figura 3.3 Estructura del arbol de directorios de un bloque.

Esta estructura se muestra en la figura 3.3.

#### **3.1.2.4.- Establecer capas.**

Las capas se usan principalmente para el *layout* del circuito integrado. El sistema Cadence usa capas para esquemas y símbolos manteniendo así una relación consistente entre la base de datos y el usuario a través de las herramientas de edición gráfica.

Cuando introducimos un objeto que es una línea, terminal o nodo el objeto es asignado automáticamente a una capa. Dependiendo de la capa en que los objetos estén colocados, el sistema interpreta que tiene un propósito específico.

A continuación veremos como las capas son definidas y editadas en el sistema Cadence y una descripción de las capas usadas para objetos esquemáticos como ejemplo.

Cadence suministra dos ficheros de capas por defecto. Uno para visualizar (*display*) y otro para dibujar (*plotting*). Puedes cambiar las definiciones de las capas en estos ficheros y guardarlos bajo un nombre diferente.

En el menú de capas que se visualiza en la parte baja de la pantalla puede observarse todas las capas que están definidas en el sistema. También posibilita el acceso al menú *Display Control* donde podremos añadir, modificar o borrar las propiedades de las capas.

Al seleccionar una capa, ésta es la única sobre la cual puedes añadir o seleccionar objetos. La capa seleccionada esta siempre indicada en la esquina superior izquierda de la pantalla. Cuando la capa *bkgnd* (*background*) esta seleccionada podremos seleccionar objetos de cualquier capa.

Veamos a continuación las capas utilizadas en un esquemático:

Solo se necesitan unas pocas capas para introducir esquemas. Además de las capas *bkgnd* , *hlite* , *jazz* , *grid1* , y *grid2* necesitamos 10 capas.

- *device*: visualiza símbolos de librería Cadence.
- *wire* (lineas): Dibuja conexiones entre pins.
- *wire* (pin): Dibuja pins en esquemas y símbolos.
- *wire* (interconexión): El sistema lo usa para visualizar la conectividad.
- *DRC*: Visualiza errores encontrados en la extracción.
- *devlab*: Visualiza las etiquetas de dispositivo usadas sobre símbolos de librería Cadence.
- *pinlab*: Etiquetas de los pins.
- *instname*: Nombre dado a un componente.
- *text*: Texto sobre un esquema.
- *border*: Visualiza líneas de carátulas.

## **3.2.- Diseñando en Cadence/Edge.**

### **3.2.1.- Configuración de la cuenta de usuario.**

Para una rápida entrada al diseño será necesario que el diseñador configure su cuenta de usuario con los ficheros necesarios.

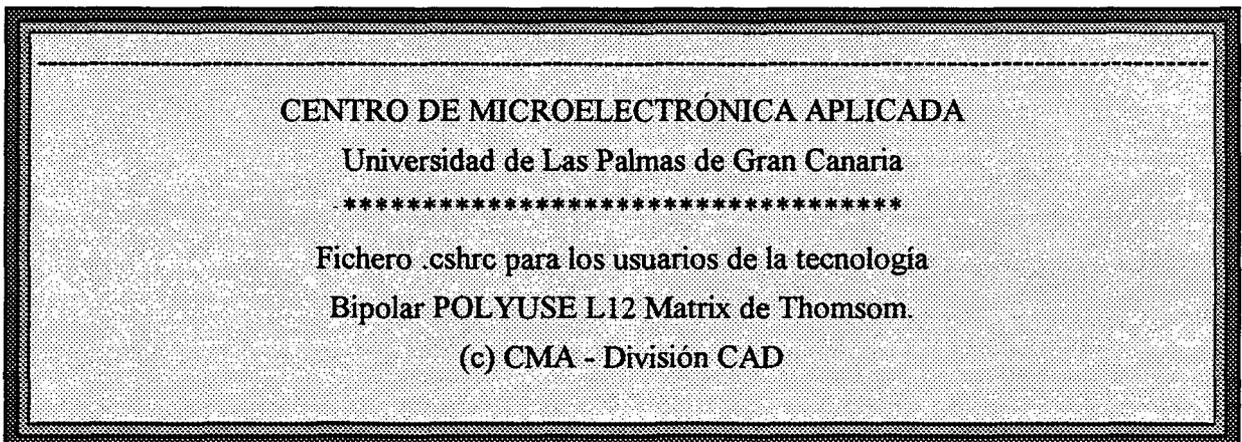
Para ello se ha creado la utilidad **setup\_user** que permite copiar los ficheros de arranque de usuario comunes en UNIX a su directorio de trabajo, creando copias de seguridad *backup* de los existentes. Será necesario invocar el comando **\$TCM\_BASE\_DIR/exe/setup\_user**.

Este comando realiza las siguientes tareas:

- \* Crea copias de seguridad de los ficheros de configuración de cshell , Cadnce/Edge y Sunview a otros con el mismo nombre y sufijo *.backup*.
- \* Copia los ficheros de configuración en el directorio raíz del usuario.
- \* Actualiza los cambios realizados. Para mayor seguridad se aconseja reinicializar su sesión de trabajo.

En el siguiente ejemplo se muestra la salida generada por el comando:

```
cmal> exe/setup_user
```



```
Las librerías se han instalado en el directorio:
>>>> "/home/soft/tcm/polyu_v2.2".
Si desea mas información ver el fichero
>>>> "/home/soft/tcm/polyu_v2.2/README"

cma1>
```

### **3.2.2.- Comienzo de un nuevo diseño.**

Para crear un nuevo diseño deberemos crear un nuevo directorio de trabajo donde se almacenen los bloques y módulos que diseñemos. Para ello se utilizará el comando `open_bip_design`.

Este comando realiza las siguientes tareas:

- Verifica si el directorio actual es el directorio de diseño (design en nuestro caso).
- Si no lo es, crea un nuevo directorio de diseño, con el nombre suministrado por el usuario, copiando en él los ficheros de configuración necesarios.
- Una vez creado el directorio de diseño, arranca el entorno Cadence/Edge en el nuevo directorio creado, y el usuario está listo para entrar en su diseño.
- Si el directorio actual es un directorio de diseño, arranca Cadence/Edge sin modificar la configuración actual.

### 3.2.3.- Continuar con un diseño existente.

Tal como se ha dicho en el apartado anterior, será posible continuar con diseño existente simplemente con invocar el comando **open\_bip\_design** en el directorio de diseño actual.

### 3.2.4.- Captura de esquemas.

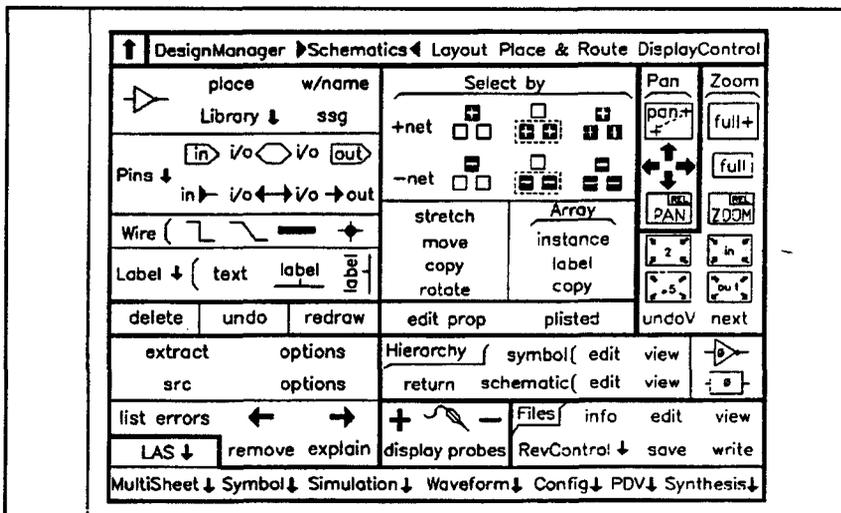


Figura 3.4 Menú para la captura de esquemas.

Para crear un esquema nuevo o editar uno ya existente debemos especificar el nombre del bloque y la representación será esquemática.

Pasos para acceder a un esquema en edición:

- 1.- Coloca el cursor en la ventana en que quieres editar el esquema.
- 2.- Visualiza el menú *schematics* que se muestra en la figura 3.4 y

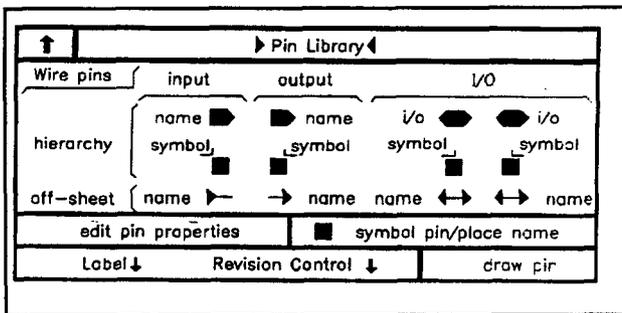


Figura 3.5 Ventana de librería de pins.

selecciona editar un fichero.

3.- Introduce el nombre del bloque.

4.- Presiona RETURN y tendrás la edición en pantalla.

A partir de este momento podremos crear nuestro circuito. Los componentes los podemos localizar en la librería de Cadence o en nuestras librerías, que previamente hemos incluido en el *path* de configuración.

Existen distintas formas de añadir un componente a nuestro esquema:

- 1.- Seleccionando la librería donde está incluido.
- 2.- Seleccionando el comando *place* y dando el nombre del componente a colocar.
- 3.- Seleccionando *place w/name*. Este comando es igual anterior pero con la posibilidad de dar un nuevo nombre al componente.
- 4.- Colocar un componente que no ha sido creado todavía. Al introducirlo con el comando *place* el sistema nos preguntará si queremos crearlo, y así lo haremos.

Las librerías de la tecnología empleada en este proyecto se encuentran en el directorio */home/soft/tcm/polyu\_v2.2*.

Podemos añadir pines de entrada y salida sobre el esquema especificando puntos de conexión eléctrica al circuito. Para ello utilizaremos el comando pin que nos mostrará la ventana de la figura 3.5.

En ella observamos una librería de pins donde debemos elegir el adecuado según su función y colocación.

También existe en el menú de esquema la posibilidad de editar los componentes de un esquema, áreas, o la totalidad y así podremos copiar, mover, rotar, borrar,...

Si deseamos ver las propiedades de un esquema debemos ejecutar el comando *plisted* para así obtener una ventana de edición de propiedades. Si lo que deseamos es ver las propiedades de un símbolo debemos ejecutar el comando *property list* después de haber

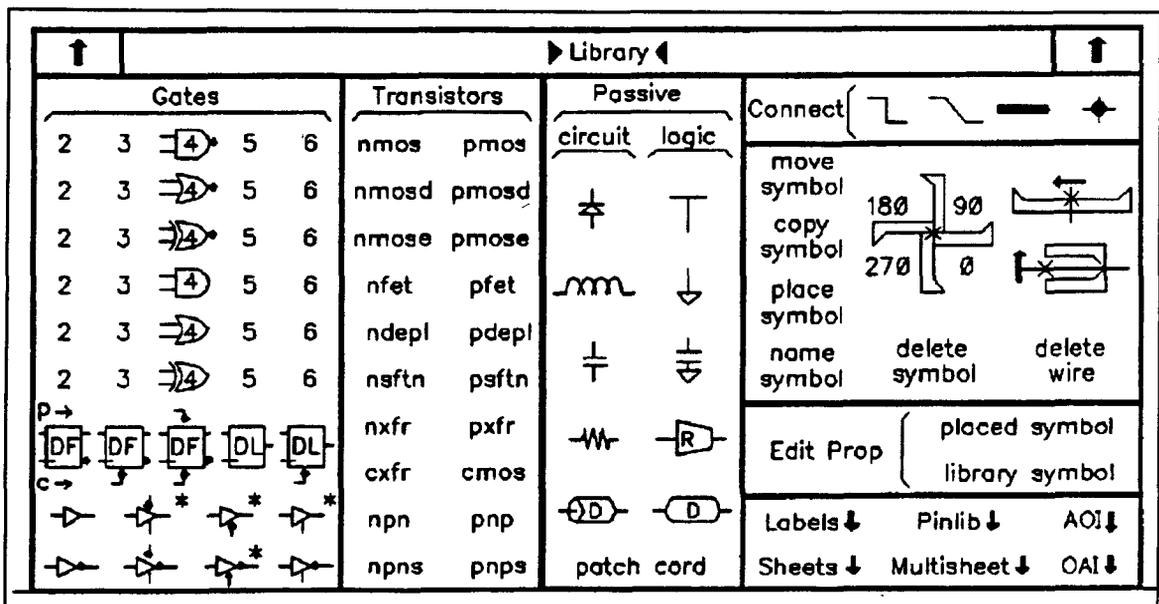


Figura 3.6 Ventana de librería.

seleccionado el símbolo del que deseamos ver las propiedades.

Las líneas de conexión del circuito las obtenemos con el comando *wire* pudiendo elegir distintos tipos de conexión. Es importante que al colocar estas uniones nos aseguremos de que llegan de punto a punto. Para ello se aconseja utilizar el comando *zoom*, que nos amplía la visión de un área, y tras esto realizar la conexión.

Podemos observar que en el menú de esquemas aparece el comando *probing nets* que son puntos de prueba con los que obtenemos información para testeo (Simulación, conexión de metal,...) del punto en cuestión. Para colocarlos utilizaremos el comando "+" y para eliminarlos el "-".

Tras haber finalizado la totalidad del esquema el siguiente paso a realizar debe ser la extracción de este esquema.

Cuando introduces un esquema con el sistema Cadence, solo introduces un dibujo de símbolos eléctricos y sus interconexiones. La información sobre la conectividad eléctrica no ha sido todavía extraída del circuito y almacenada en la base de datos.

Una de las ventajas de la captura de esquemas electrónicos es que el esquema puede servir como entrada a otras herramientas de diseño como puede ser la simulación.

Si deseamos usar el esquema como entrada de otras herramientas debemos

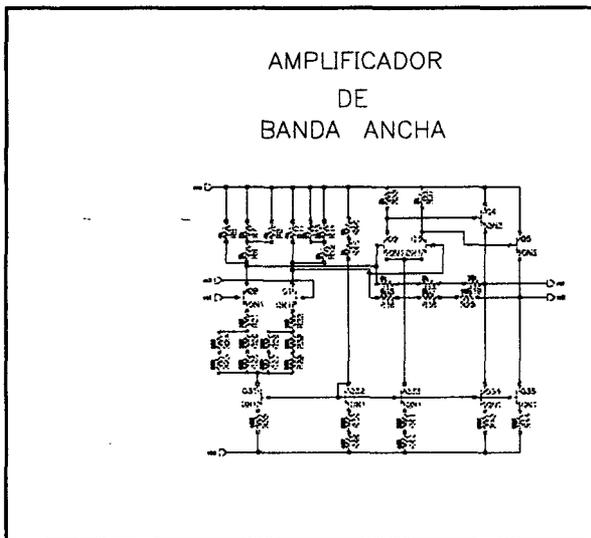


Figura 3.7 Esquemático de un circuito.

extraer la conectividad del esquemático cuando hayamos terminado de crearlo.

La extracción esquemática se realiza seleccionando el comando *extract* sobre el menú de esquema.

Durante la extracción de la conectividad eléctrica, el programa de

extracción comprueba algunas reglas simples de chequeo. Si detecta alguna violación de estas reglas durante la extracción lo notifica haciendo parpadear el pin, línea, componente,... en la capa DRC.

Debemos corregirlos y volver a realizar la extracción. Tras una correcta extracción debemos "grabar el esquema".

"La extracción del esquemático no es jerárquica". Esto significa que debemos extraer el esquema de cada nivel de jerarquía.

Para introducir jerarquía en esquemáticos reemplazaremos subcircuitos enteros por símbolos de éstos incluyendo sus propiedades y conexiones eléctricas.

Como ejemplo de jerarquía podemos observar la figura 3.7 con el esquemático

y la figura 3.8 con la representación símbolo.

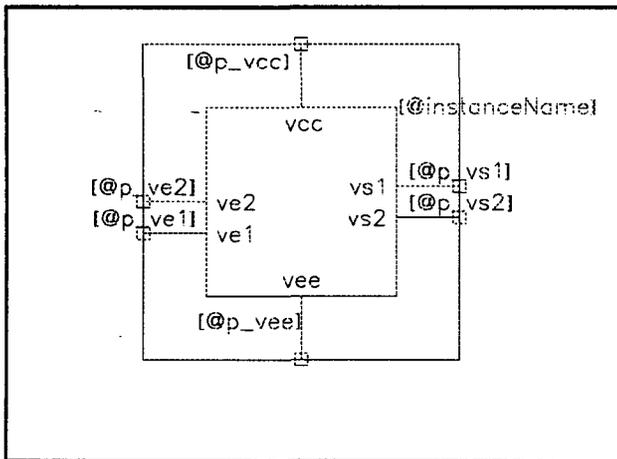


Figura 3.8 Símbolo del circuito.

Podemos usar herramientas Cadence para la creación automática de símbolos ó usar el entorno de edición interactiva par crear símbolos a nuestra medida.

La generación automática se realiza con los comandos TSG y SSG. Para hacerlo manualmente usaremos el menú *Draw Symbols*.

En el diseño realizado utilizaremos la generación automática. Para ello Cadence suministra dos programas de fácil uso para la creación de símbolos. El generador de texto a símbolo (TSG) y el generador de esquemático a símbolo (SSG).

El TSG acepta como entrada un fichero de texto en el cual uno o más símbolos son descritos , y automáticamente genera una representación para cada símbolo descrito.

El SSG acepta como entrada un esquemático que contiene como mínimo un pin y genera un símbolo del esquemático. Éste es el usado para generar el símbolo de la figura 3.8. Debido a que en este proyecto se han analizado los esquemas de los subcircuitos utilizaremos siempre el SSG ya que disponemos de los esquemáticos de las células.

### **3.2.5.-Generación de *netlist*.**

Cuando creas o editas un esquema, los datos del esquema son propiedad de la base de datos del Cadence. La base de datos del Cadence usa datos en formato binario.

Si deseamos utilizar estos datos en herramientas que no utilizan la estructura de la base de datos del Cadence necesitas obtener información de la base y generar el formato requerido para la herramienta que vas a utilizar.

El sistema Cadence dispone de la capacidad de generar descripciones textuales de la conectividad del bloque y sus propiedades. Estas descripciones textuales se llaman *netlist*. El proceso para generar *netlist* se llama *netlisting* y las herramientas que los generan *netlisters*.

Cadence suministra la posibilidad de la generación automática de una serie de formatos de *netlist*. Por ejemplo puedes generar *netlist* para los formatos utilizados por los simuladores SILO , HILO™, Verilog, SPICE™, HSPICE™. Además puedes crear tu propio formato de *netlist*.

Cadence dispone de un completo entorno donde los simuladores pueden ser integrados dentro del sistema Cadence presentando un consistente interface de usuario. Este entorno se llama Interface de Simulación (SI) el cual controla también los generadores de *netlist*.

Una vez el *netlist* es generado puede ser cargado en el simulador y proceder a su simulación.

Un ejemplo de *netlist* se muestra a continuación:

```
* net 0 = gnd!
* net 1 = /vin3
* net 2 = /vin4
* net 3 = /vin1
* net 4 = /vcc-
* net 5 = /Q0.E
* net 6 = /Q2.E
* net 7 = /Q3.E
* net 8 = /Q5.E
* net 9 = /Q6.E
* net 10 = /R10.PLUS
* net 11 = /R10.MINUS
* net 12 = /Q6.B
* net 13 = /R11.MINUS
* net 14 = /Q8.E
* net 15 = /R13.PLUS
* net 16 = /R15.PLUS
* net 17 = /Q7.E
* net 18 = /R17.PLUS
* net 19 = /vs2
* net 20 = /vs1
* net 21 = /vcc+
* resistor(2) = /R18
R$#2 4 18 100
* resistor(3) = /R17
R$#3 18 17 400
* resistor(4) = /R16
R$#4 4 16 100
* resistor(5) = /R15
R$#5 16 9 400
* resistor(6) = /R14
```

```
R$#6 15 4 100
* resistor(7) = /R13
R$#7 15 14 400
* resistor(8) = /R9
R$#8 13 12 100
* resistor(9) = /R11
R$#9 11 13 200
* resistor(10) = /R10
R$#10 10 11 200
* resistor(11) = /R12
R$#11 0 10 1k
* resistor(12) = /R31
R$#12 20 21 1k1
* resistor(13) = /R30
R$#13 19 21 1k1
* resistor(14) = /R29
R$#14 6 8 300
* qn1(15) = /Q8
Q15 12 12 14 N1
* qn1(16) = /Q7
Q16 8 12 17 N1
* qn1(17) = /Q6
Q17 6 12 9 N1
* qn1(18) = /Q5
Q18 7 0 8 N1
* qn1(19) = /Q4
Q19 20 1 7 N1
* qn1(20) = /Q3
Q20 19 2 7 N1
* qn1(21) = /Q2
Q21 5 3 6 N1
```

```
* qn1(22) = /Q1
Q22 20 2 5 NI
```

Este *netlist* es el resultado de la generación para el simulador SPICE de un fichero de entrada. Observamos como describe la conectividad de las resistencias y distintos transistores, así como el tipo de transistor del que se trata.

### 3.2.6.- Simulación.

Tras la realización del esquema del circuito debemos comprobar que efectivamente realiza la función para la que ha sido diseñado. Para ello debemos simular el circuito.

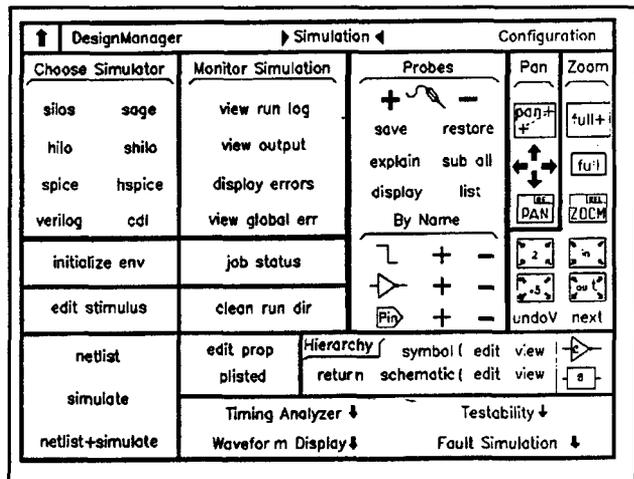


Figura 3.9 Menú de simulación.

La casa Thomson suministra un *software* junto con la tecnología llamado Polytool, basado en el simulador Spice y el programa de diseño Orcad. Las especificaciones del manual nos indican que con esta herramienta podremos realizar todo el diseño, desde un circuito esquemático hasta la realización del *layout*. Al no disponer de esta herramienta se ha optado por la simulación dentro del Cadence.

En el entorno Cadence Edge disponemos de un interface de simulación para

distintos simuladores. Desplegando el menú de simulación nos encontramos con la opción de elegir cualquiera de los programas de simulación que nos muestra. Debido a que el fabricante nos suministra las librerías para Spice, es éste el simulador que en un principio se utilizará.

### **3.2.6.1.-Simulador Spice**

Spice es un *software* de simulación de circuitos para análisis transitorios, de continua y de alterna para circuitos lineales. Este *software* ha sido diseñado por la Universidad de California. Thomson suministra una librería de interfaces para Spice versión 2G.6. Thomson no vende el programa Spice pero suministra una copia de dominio público del programa y su manual, libre de cargo, cuando compras el interface.

Los circuitos podran contener resistencias, condensadores, bobinas, fuentes independientes de tensión y corriente, varios tipos de fuentes dependientes, líneas de transmisión y los cuatro tipos mas comunes de semiconductores: Diodos, BJT's, JFET's y MOSFET's.

El Spice dispone de modelos para dispositivos semiconductores , y el usuario solo necesita especificar los valores de los parámetros del dispositivo a utilizar. El modelo para el transistor bipolar está basado en el modelo Gummel-Poon. Si los parámetros para Gummel-Poon no están especificados, usará el modelo de Ebers-Moll.

Spice no tiene un comando para leer un fichero de entrada, así que el fichero

*netlist* debe ser incluido en el fichero *control* junto con el de entorno de simulación. Otros comandos o el fichero de estímulos también pueden ser incluidos. Además del fichero *netlist* aparecen también los ficheros *spice.inp* y *spice.sim* usados para los datos del estímulo y comandos del simulador respectivamente. Al ejecutar en la ventana de simulación el comando *initialize.env* se crean estos ficheros automáticamente en un nuevo directorio. El nombre de este directorio, así como el nombre del esquema del que queremos obtener el fichero *netlist* nos lo preguntan al ejecutar el comando de inicialización de entorno de simulación.

Para poder simular el circuito necesitamos además del fichero *netlist* una serie de estímulos los cuales podemos incluirlos directamente en el fichero *netlist* o en el fichero *spice.inp*.

El lenguaje de simulación y test (STL) genera *spice.inp* y *spice.sim* automáticamente.

#### FICHERO "CONTROL"

```
.temp 25
.options acct opts nopage nomod itl5=0 limpts=10000
+ gmin=1e-10 reltol=1e-3 vntol=1e-3 abstol=1e-9 chgtol=1e-12
+ defas=12e-12 defad=12e-12
.width in=80 out=80

[!/home/soft/tcm/polyu_v2.2/SPICE_UTIL/polyu_simple.spc]
[!spice.inp]
[!spice.sim]
[!netlist]* Simplified POLYUSE BIPOLAR parameters, 25C
```

```
* *****  
* ** typical-case simulation **  
* *****  
*  
* gmin=1e-10  should avoid dc convergence problems  
* vntol=1e-3      "  
* abstol=1e-9     "  
* reltol=1e-3    improves the accuracy, may be increased  
end
```

Los tipos de análisis que podemos hacer con el Spice son los siguientes:

-Análisis en continua: Determina el punto de operación en continua del circuito con bobinas cortocircuitadas y condensadores abiertos. El análisis en continua se realiza automáticamente antes de un análisis transitorio o uno en alterna.

-Análisis en alterna para pequeña señal: El análisis en alterna lineales para pequeña señal calcula las variables de salida en función de la frecuencia.

-Análisis transitorio: Calcula las variables de salida en función del tiempo en un intervalo especificado por el usuario.

-Análisis a diferentes temperaturas: Todos los datos de entrada para Spice han sido medidos a 27°C. La simulación también asume una temperatura nominal de 27°C. El circuito puede ser simulado a otras temperaturas usando el comando .TEMP.

Las librerías necesarias con los modelos de los dispositivos a utilizar las tenemos disponibles en el software suministrado por el fabricante, las cuales mostramos a

continuación. El significado de cada parámetro lo podemos encontrar en el apéndice que a tal efecto se ha incluido en esta memoria. Disponemos además de los parámetros típicos, que son los mostrados, una serie de parámetros máximos y mínimos que nos suministran en la documentación de la tecnología.

**FICHERO DE MODELOS DE DISPOSITIVOS**

\* MODELES POUR POLYUSE L FICHER REDUIT POUR TPENSERG 270290

\* MODEL FOR SMALL SIZE NPN N1

.MODEL N1 NPN

+IS=1.8E-16 BF=130 NF=1 VAF=35 IKF=11.5E-3

+ISE=0.25E-15 NE=1.6 BR=1.8 NR=1 VAR=5 IKR=6E-3

+ISC=0 NC=2 RB=300 IRB=1.5E-4 RBM=65 RE=7 RC=140

+CJE=140E-15 VJE=.85 MJE=.4 TF=47E-12 XTF=1 ITF=5E-3

+CJC=170E-15 VJC=.6 MJC=38 XCJC=2 TR=19E-9

+CJS=360E-15 VJS=.67 MJS=.43

+XTB=2.5 EG=1.205 XTI=2.9

+AF=1.33 KF=1E-14

\*

\* MODEL FOR MEDIUM SIZE NPN N2

.MODEL N2 NPN

+IS=3.6E-16 BF=130 NF=1 VAF=35 IKF=23E-3

+ISE=0.5E-15 NE=1.6 BR=2.9 NR=1 VAR=5 IKR=9.4E-3

+ISC=0 NC=2 RB=160 IRB=3E-4 RBM=32 RE=4 RC=50

+CJE=280E-15 VJE=.85 MJE=.4 TF=47E-12 XTF=1 ITF=10E-3

+CJC=340E-15 VJC=.6 MJC=38 XCJC=25 TR=15E-9

+CJS=460E-15 VJS=.67 MJS=.43

+XTB=2.5 EG=1.205 XTI=2.9

+AF=1.33 KF=5E-15

\*

\* MODEL FOR LARGE SIZE NPN N4

.MODEL N4 NPN

+IS=9E-16 BF=130 NF=1 VAF=35 IKF=57.3E-3  
 +ISE=1.25E-15 NE=1.6 BR=2.4 NR=1 VAR=5 IKR=12.5E-3  
 +ISC=0 NC=2 RB=60 IRB=7.5E-4 RBM=13 RE=2.5 RC=30  
 +CJE=700E-15 VJE=.85 MJE=.4 TF=47E-12 XTF=1 ITF=25E-3  
 +CJC=660E-15 VJC=.6 MJC=.38 XCJC=.25 TR=18E-9  
 +CJS=840E-15 VJS=.67 MJS=.43  
 +XTB=2.5 EG=1.205 XTI=2.9

\*  
 \* MODEL FOR LATERAL PNP PL  
 MODEL PL PNP

+IS=.58E-16 BF=71 NF=1 VAF=23 IKF=59E-6  
 +ISE=1.1E-16 NE=1.52 BR=10 NR=1 VAR=18  
 +IKR=30E-6 ISC=57E-18 NC=1.77 RB=365 IRB=1.2E-6  
 +RBM=25 RE=9 RC=8 CJE=11E-15 VJE=.65  
 +MJE=.25 TF=16E-9 XTF=1 ITF=550E-6  
 +PTF=9 CJC=190E-15 VJC=.67 MJC=.43 XCJC=1  
 +TR=90E-9 XTB=2.2  
 +EG=1.205 XTI=2.6 KF=0 AF=0 FC=.95

\*  
 \* MODEL FOR VERTICAL SUBSTRATE PNP PS  
 \*  
 MODEL PS PNP

+IS=2E-14 BF=300 NF=1 VAF=36 IKF=13E-3  
 +ISE=2.22E-16 NE=1.2 BR=1 NR=1 VAR=12  
 +IKR=1.3E-3 ISC=0 NC=2 RB=270 IRB=300E-6  
 +RBM=150 RE=1E-2 RC=30 CJE=1E-12 VJE=.67  
 +MJE=.43 TF=3.5E-9 XTF=1 ITF=300E-3  
 +PTF=0 CJC=2.2E-12 VJC=.5 MJC=.3 XCJC=1  
 +TR=0.5E-6 XTB=2.2  
 +EG=1.11 XTI=3 FC=.5

\*  
 \*

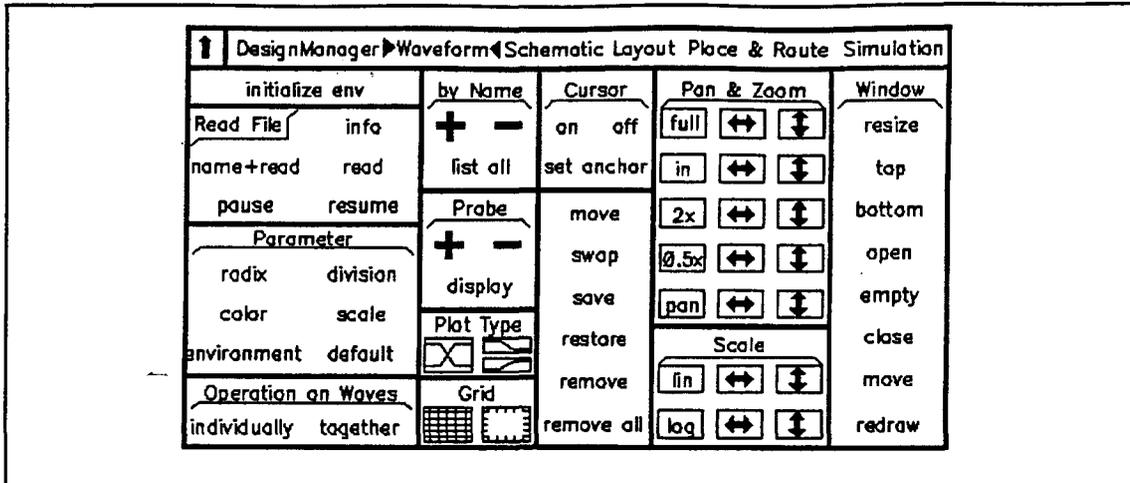


Figura 3.10 Menú del Waveform Display.

### 3.2.7.-Visualización de formas de onda.

Tras haber realizado la simulación del circuito es necesario una herramienta que nos permita comprobar que la función que realiza éste es la correcta. Para ello están las herramientas de visualización gráfica de formas de onda.

El sistema Cadence nos proporciona la herramienta Waveform Display. Esta herramienta es un visualizador de formas de onda cuyos ficheros son escritos en formato WFS. Estos ficheros pueden ser de texto o binarios. Cadence los genera en binario ya que son más compactos y fáciles de procesar.

Para acceder a esta herramienta debemos desplegar el menú que con su nombre está en las ventanas de simulación, o desde cualquier otra ventana principal. El primer paso a realizar siempre es la inicialización del entorno ejecutando el comando *initialize env*. Esto

se debe a que si no se hace así el programa no sabe a que diseño nos estamos refiriendo, y por lo tanto no sabe a que ficheros se debe remitir a la hora de visualizar una onda, por lo que daría un error.

Para visualizar una forma de onda hay distintos procedimientos. Podemos ver las formas de onda de los nodos marcados con *probe* simplemente con ejecutar *read* despues de inicializar el entorno. Otra forma es sacar una lista de las formas de onda que podemos observar utilizando el comando *list* y asi elegir la que queramos. Para añadir nuevos nodos debemos situarnos en la ventana del esquema y marcarlos con el comando *probe*. Seguidamente ejecutaremos *read* y se incluiran estos nuevas formas de onda de los nodos en la pantalla.

Esta herramienta dispone de comandos de edición como pueden ser el zoom, mover, borrar, seleccionar, ...Ademas podemos seleccionar si vamos a visualizar las formas de onda en una sola ventana o en varias segun deseemos. La opción del fondo enrejado o nó esta tambiem disponible. Podemos utilizar cursores para realizar medidas sobre las formas de onda.

Tras el trabajo realizado sobre esta herramienta se ha comprobado que tiene dos limitaciones importantes a la hora de simulaciones en el campo analógico principalmente.

Estas son:

- 1.- El Waveform Display no está diseñado para manejar múltiples análisis de simulaciones. Solo podrá manejar un análisis por simulación.
- 2.- El Waveform Display no visualiza los resultados de analisis en

alterna para pequeña señal. Esto significa que no podemos visualizar respuestas en frecuencia, cuestión fundamental en el diseño analógico. La única forma de obtener una idea de como son estas formas de onda es utilizando los comandos de Spice *.print* o *.plot* para generar salidas de texto y leerlo usando el comando *view output*.

Estas limitaciones van a hacer que no realicemos todo el proceso de diseño dentro de entorno Cadence, sino que tendremos que utilizar *software* exterior para la realización de éste.

### 3.2.8.- Realización del *layout*.

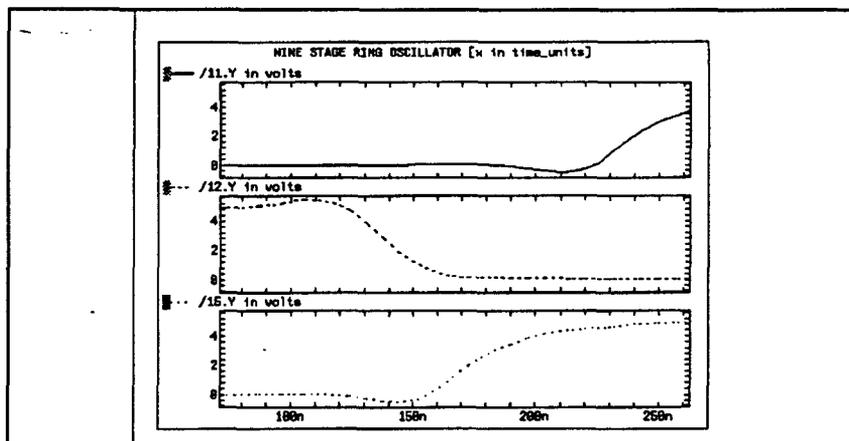


Figura 3.11 Pantalla de visualización de ondas.

Tras simular y verificar la función del circuito debemos comenzar con el proceso del trazado físico del diseño. Para ello el sistema Cadence nos suministra un interface que podemos utilizar desplegando el menú de *layout*.

Al ejecutar el comando editar un fichero, obtendremos el modelo L1200 de los distintos tipos de encapsulados disponibles en la tecnología. En nuestro caso no tenemos la opción de elegir el tipo ya que solo se ha suministrado el modelo L1200 anteriormente indicado. En él podemos observar la distribución en tiles de los componenetes disponibles.

Usando el editor de layout podemos dibujar, células básicas, jerarquías, diseños completos. En el caso que nos ocupa solo dibujaremos las conexiones y los contactos, con sus correspondientes capas de metal, para obtener nuestro circuito.

Cuando archivemos este diseño quedará salvado en el directorio */propietario/nombre\_diseño/layout* con el nombre *current*.

Se debe incluir en el *blockpath* de la configuración la librería *schema* que es necesaria para la sesión de edición del *layout*.

```
blockpath="/cds/etc/sdalib/schema/  
/cds/etc/sdalib/schema/sheets/38ths"
```

En el menú de *layout* se pueden encontrar distintos tipos de comandos que podemos dividir en los siguientes apartados:

- 1.- Comandos de entrada de datos.
- 2.- Comandos de edición.
- 3.- Comandos DRC.
- 4.- Comandos de jerarquía.

5.-Comandos de regla.

6.- Otros comandos.

1.- Dentro de este tipo de comandos tenemos:

*Shapes*. Contienen iconos que son comandos gráficos para crear líneas, pads, rectángulos, polígonos, círculos.

*Conics*. Menu desde el cual puedes seleccionar secciones cónicas para crear formas geométricas especiales.

*Label*. Menú desde el cual puedes cambiar el estilo, el tamaño, la justificación,...

*Place label*. Coloca etiquetas de texto.

*Place cell*. Coloca una célula que anteriormente se ha especificado.

*Array*. Menu para crear *arrays* de mayor complejidad.

2.- Dentro de este tipo de comandos tenemos el mover, copiar, mover una capa, copiar a una capa, borrar, corta,... y en general los comandos típicos de un menú de edición.

3.- Los comandos de chequeo los vamos a utilizar desplegando el menú de Verificación del Diseño Físico (PDV, *Physical Design Verification*). Nos vamos a encontrar tres comprobaciones fundamentales:

Comprobación de las reglas de diseño (DRC): La sección DRC dentro del menú PDV es la que podemos observar en la figura. El DRC se encarga de comprobar los posibles errores de diseño que hemos cometido. Tiene diferentes modos de operación: el *full flat*, que chequea el diseño como si no existieran jerarquías, el *full hierarchical* sin optimización que chequea jerárquicamente, chequeando cada bloque en todos los niveles de jerarquía, y por último, el *full hierarchical* que es igual que el anterior solo que cada bloque se chequea solo la primera vez.

Antes de ejecutar el DRC debemos especificar las reglas de diseño por las que se ha de guiar. Utilizando el comando *specific rules files* (especificar reglas de diseño) introduciremos nuestro fichero de reglas. La tecnología nos suministra dos posibles tipos de ficheros de reglas de diseño, el *polyuse104* y el *polyuse105*.

Otros comandos del DRC son la visualización de errores, que realiza un *zoom* de los errores uno por uno, o el *explain* (explicación) que nos da la información sobre por qué se ha producido el error.

Extracción del Diseño Físico (Pdextract). El propósito del Pdextract es generar una representación de *layout*, la cual contendrá su funcionalidad eléctrica (incluyendo elementos parásitos). La representación resultante puede ser usada para comparaciones, simulaciones, o algún otro proceso típico de los *extracted*. Tiene los mismos tipos de funcionamiento que el DRC. Antes de la extracción debemos indicar el fichero a usar. El suministrado por la tecnología es el *xtr*.

Comparación del Diseño Físico (Pdcompare). Este comando genera un *netlist* del fichero *extracted* y otro del fichero esquemático para posteriormente realizar una comparación de ellos. Automáticamente genera el subdirectorio *comprum* tanto en el *layout* como en el esquemático donde quedarán almacenados los ficheros generados.

4.- Los comandos de jerarquía son aquellos que utilizaremos para movernos a través de los distintos niveles de jerarquía y realizar ediciones de éstas. Los comandos son los de *edit*, *return*, *pushview*,...

5.- Los comandos para la utilización de la regla son los que hacen que la regla se visualice en pantalla y podamos mover los cursores para realizar medidas.

6.- Dentro de este tipo se incluyen comandos para diferentes funciones no básicas como pueden ser los zoom, las panorámicas,...

### **3.2.9.- Herramientas exteriores al Cadence.**

Debido a que la herramienta *Waveform Display* de Cadence, no es lo suficientemente potente se ha optado por realizar las simulaciones en Hspice, y así poder visualizar los resultados con la herramienta GSI. Para ello se utilizará el *software* del simulador Hspice que disponemos, independiente de Cadence, ya que la utilización de éste dentro del sistema Cadence no es igual de potente. Esto es debido a una serie de razones que se enumeran a continuación:

1.- Los ficheros de formas de onda generados por Hspice dentro de Cadence son demasiados extensos, provocando interrupciones del sistema al faltar espacio de memoria en el disco asignado.

2.- Las librerías que suministra el fabricante son específicas para Spice, por lo que no serán compatibles para Hspice.

3.- Seguiríamos teniendo el mismo problema a la hora de visualizar las señales, ya que la única herramienta de visualización dentro del Cadence es el *Waveform Display*.

### **3.2.9.1.- Simulador HSPICE.**

Es un simulador de propósito general. Tiene un extenso juego de modelos de dispositivos, incluyendo modelos para pequeña geometría Mosfet's y Mesfet's. El programa se diseñó para ser compatible con formatos de entrada Spice y Hsing.

Cadence dispone de un completo interface de Hspice, pero trabajar con este supondría incompatibilidad de librerías, ficheros de simulación demasiado extensos y visualización en *Waveform* limitada. Por ello, se utilizará el *software* de Hspice independiente de Cadence.

Las principales características que diferencian al Hspice del Spice son:

- Convergencia superior.
- Inclusión de HSPLIT y GSI para la visualización de gráficas.
- Referencia y nombramiento de nodos jerárquica.

Para realizar una simulación en Hspice necesitamos un fichero de entrada y una librería de modelos. El fichero de entrada es prácticamente igual al de Spice. Solo habrá que realizar unas pequeñas variaciones que describiremos en el capítulo dedicado a la simulación (Capítulo IV). La librería de modelos la localizamos en los directorios que nos suministra la empresa, junto con la tecnología. Estos modelos son totalmente compatibles con los de Hspice.

Para ejecutar una simulación Hspice simplemente debemos ejecutar Hspice mas el nombre del fichero. Ejemplo: **hspice fichero\_entrada**.

Se dispone de la facilidad de utilizar el software en cualquier máquina de la red ya que posee licencia flotante.

### **3.2.9.2.- GSI. Interface de Simulación Gráfico.**

El GSI, junto con el HSPLIT y el comando de Hspice .GRAPH son herramientas gráficas para poder ver, medir o hacer impresiones de los resultados de una simulación Hspice. Estas herramientas pueden representar voltaje de nodos, corrientes de elementos, respuesta del circuito y expresiones algebraicas.

El .GRAPH colocado en el fichero de entrada produce automáticamente, cuando Hspice se ejecuta, una copia de las gráficas de los nodos especificados.

El GSI y HSPLIT son interactivos. El GSI dispone de más características que HSPLIT.

GSI está basado en la herramienta xWindows. Debe tener un entorno xWindows como el openWindows ó Motif ejecutándose en la máquina. Si no se posee xWindows habrá que ejecutar el HSPLIT.

El visualizador utilizado en la realización del trabajo es el GSI. Alguna de las características principales que posee son las siguientes:

- .- Gran flexibilidad.
- .- Formato de fichero independiente de la máquina.
- .- Configuración propia.
- .- Posibilidad de utilización de expresiones algebraicas.
- .- Zoom x, y.
- .- Variedad de análisis.
- .- Utilización de cajas de elementos de diálogo.
- .- Lectura de múltiples ficheros de gráficos para visualizar curvas simultaneamente.
- .- Especificaciones flexibles para ventana de gráficos (color del título, nodos, etiquetas y reglas), paneles (tipos de escala, símbolo de frecuencias, grids, ...) y propiedades de las curvas (estilo de línea, color y símbolos).
- .- Especificaciones de título para impresión.

Los tipos de ficheros generados poseen una extensión en concreto según el análisis que contengan, así:

fichero.**tr0**.....Análisis transitorio

fichero.**ac0**.....Análisis AC

fichero.**sw0**.....Análisis DC

La opción "*options post*" debe ser colocada en el fichero de entrada. Ésta se encarga de crear el fichero de datos gráficos y carga los resultados de la simulación para GSI o HSPLIT.

El comando "*protect*" limita el número de curvas cargadas.

El comando "*interp*" limita el número de puntos cargados.

Por último comentar que una de las mayores ventajas del GSI es el acceso al *software* desde una máquina remota. Para ello se ejecutará los siguientes comandos desde una ventana de comandos del openWindows:

**xhost +**

**rlogin máquina** (máquina que posee el GSI)

**setenv DISPLAY máquina\_usada:0** (para visualizar en nuestra máquina)

**gsi**

La ejecución de GSI se puede realizar de tres formas distintas. La primera ejecutando *gsi*. La segunda ejecutando *gsi* más el nombre del fichero que queremos cargar.

La tercera ejecutando `gsi -a`, cargando de esta forma las últimas gráficas visualizadas.

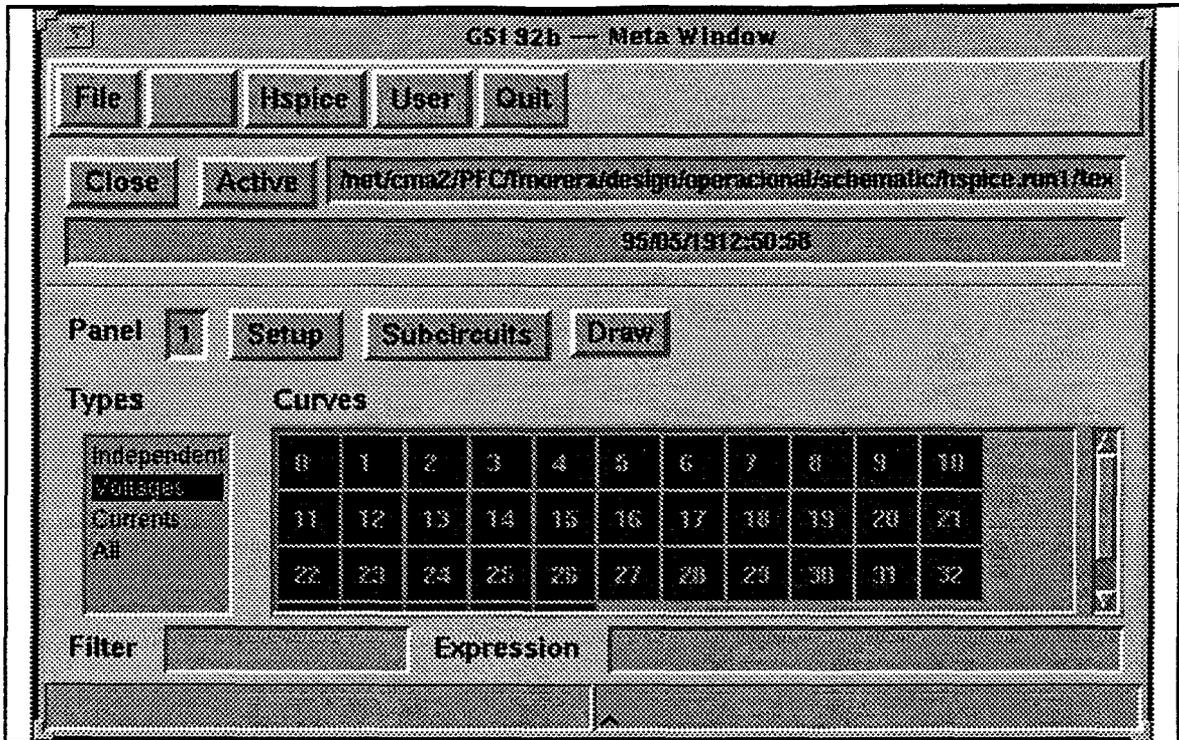


Figura 3.12.- Cuadro de diálogo para control de la visualización.

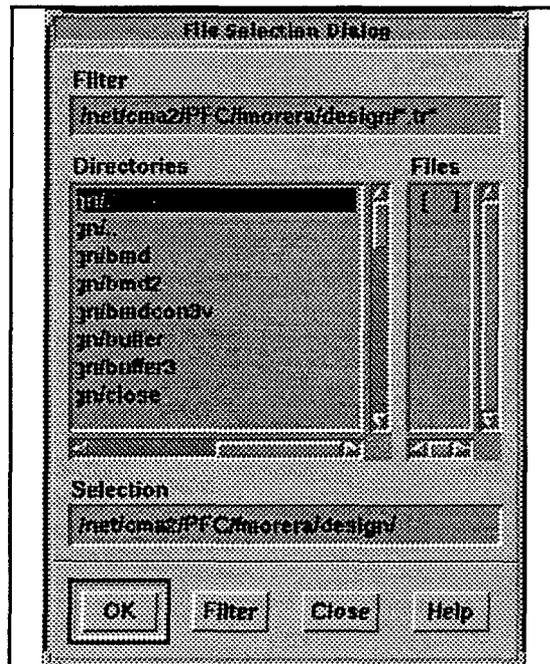


Figura 3.13.- Cuadro de diálogo para la selección de fichero.

# Capítulo IV

## Ruta de Diseño

El propósito de este capítulo es la descripción del proceso seguido para el diseño de circuitos con la tecnología bipolar suministrada por Thomsom. Todas las conclusiones obtenidas y las decisiones tomadas son fruto de horas de pruebas y consultas tanto a profesores experimentados en la materia como a libros especializados y manuales de usuarios. No se descarta ninguna otra posibilidad , es más, probablemente se pueda realizar de otra manera, cuestión que dejamos pendiente a posteriores estudios. Por el momento se ha logrado obtener una ruta de diseño válida y bastante potente que es parte fundamental de este proyecto. La ruta de diseño obtenida es la que se desglosa en los siguientes puntos:

- .- Realización de los esquemas en Cadence.
- .- Generación de los *netlist* para Spice en Cadence.
- .- Modificación de los *netlist* para Spice.
- .- Simulación en Hspice.
- .- Visualización de resultados en GSI.
- .- Realización del *layout* en Cadence.
- .- Verificación del Diseño y encapsulado.

Describiremos todos los pasos seguidos durante el diseño y verificación de la tecnología tanto los definitivos como los pasos intermedios que se han suprimido, y sus causas. Se realizará de esta forma ya que la memoria de este proyecto está enfocada a ser una obra de consulta a posteriores diseñadores.

Thomsom suministra un software llamado Polytool basado en el simulador Spice y el programa de diseño Orcad. Según las especificaciones descritas en el manual de la tecnología, observamos que con esta herramienta podremos realizar todo el proceso de diseño, desde el circuito esquemático a la realización del layout. Al no disponer de esta herramienta la ruta de diseño planteada inicialmente fue la siguiente:

Primero se realizarían las simulaciones sobre PC's, usando el simulador Pspice (usado en Polytool), y así comprobar de una manera mas cómoda el correcto funcionamiento de la tecnología. Una vez simulado el diseño, se realizarán las simulaciones en las estaciones de trabajo con la certeza del funcionamiento correcto, y así disponer de simulaciones mas completas con Spice (mucho mas potente que su versión para PC's). Este paso se realizará dentro del entorno Cadence Edge donde previamente habremos introducidos los esquemas. Una vez simulado, y comprobado su perfecto funcionamiento se realizaría el *layout* del circuito para su posterior verificación.

Esta ruta sufrirá algunos cambios conforme se presente las limitaciones en algunas de las herramientas (o mejores prestaciones en otras), lo cual iremos describiendo paso a paso y modificando según nuestra conveniencia. Así introduciremos las simulaciones

---

en Hspice y las visualizaciones de formas de onda en GSI, exterior al sistema Cadence.

Describamos como se ha obtenido la ruta de diseño definitiva:

#### **4.1.-Simulaciones en Pspice.**

El Pspice es un *software* de simulación de circuito electrónicos para análisis transitorio, de continua y en alterna para circuitos lineales. Utiliza el modelo Gummel & Pool para la simulación de dispositivos semiconductores, aunque si no se le suministra los parámetros suficientes lo realizará con el modelo de Ebers Mol.

Para ejecutar una simulación en Pspice debo disponer de un fichero de entrada y de las librerías de los dispositivos que se van a utilizar.

En Pspice el fichero de entrada es el que nos describe el circuito. Para ello en un editor de texto se introducen los elementos del circuito y los nodos a los que estan conectados. Además dentro de este fichero debemos incluir cual es la librería a utilizar. Lo indicado anteriormente es lo mínimo para describir un circuito en Pspice, conforme realicemos circuitos mas complicados será conveniente incluir subcircuitos (.SUBCKT) , otros archivos (.INCLUDE) ,...

Un circuito simple lo podemos introducir fácilmente de forma directa en el editor de texto, pero cuando el circuito crece en el número de dispositivos sería muy trabajoso

y nos llevaría mucho tiempo. Usando la utilidad NETLIST del Orcad hemos realizado un pequeño programa .bat que construye el circuito en formato Spice a partir de un esquema dibujado en el programa Draft del Orcad. El listado de este programa lo tenemos a continuación:

```
@echo off
cls
echo
echo Sistema automático de generación de netlists.
echo.
CHOICE /C:GT /TT,5 ¿ Editor de gráficos o de textos
if errorlevel 2 goto fed
***** Ir al subdirectorio OrCAD
cd cad
echo.
echo Llamando al OrCad ...
draft %1.drt >nul
echo.
echo Reanotando dispositivos ...
annotate %1.drt /u /m >nul
echo.
echo.
echo Resolviendo netlist ...
netlist %1.drt %1.spi /s spice /n >nul
***** Volver al subdirectorio spice
echo.
if exist d:%1.spi echo Generado %1.spi
move d:%1.spi . >nul
echo.
echo Construyendo %1.inc ...
echo ***** SIMULACION DE %1 ***** > cab.$$$
copy cab.$$$+inc.spi+%1.spi %1.inc >nul
del cab.$$$ >nul
if exist preedit.bat call preedit.bat
echo Llamando al editor...
```

```
edit %1 inc
echo Edición completada.
if exist postedit.bat call postedit.bat
goto fin
:fed
    copy %1.cir %1.$$$ > nul
    call dobak %1.cir
    move %1.$$$ %1.cir > nul
    edit %1.cir
:fin
```

Con el Pspice se han realizado análisis transitorios, alterna, y continua de los diferentes bloques del diseño realizado, incluyendo otros bloques para estudio de la tecnología. Usando el procesador de gráficos PROBE que incluye el *software* de Pspice visualizamos esta respuestas y realizamos las modificaciones correspondientes hasta lograr los resultados satisfactorios.

La limitación que nos encontramos al utilizar este simulador es que tenemos disponible el Pspice 4.04 y éste a la hora de intentar simular un circuito de mayor magnitud no realiza la simulación, ya que no admite demasiada cantidad de transistores. Por tanto el Pspice nos va a servir para realizar las comprobaciones previas de las celulas del diseño por separado, pero nunca para la comprobación del circuito completo. En caso de necesidad, por imposibilidad de acceso a otro *software* de mayor potencia, se realizaría una simulación por partes, pero como no es nuestro cas, el siguiente paso sería simular en las estaciones de trabajo con el simulador Spice.

## 4.2.-Simulaciones en Spice.

Dentro del entorno de diseño de circuitos electrónicos Cadence Edge disponible en el laboratorio del CMA nos encontramos con el simulador analógico Spice.

Para empezar a utilizar las estaciones el primer paso a dar, es conocer una base de las órdenes Unix mas comunes para poder moverse dentro de este sistema y asi llevar a cabo con comodidad el proceso de diseño y simulación.

Se ha creado un fichero llamado **open\_bip\_design** mediante el cual nos situa en la ventana de diseño de Cadence Edge ya preparados para comenzar. El proceso seguido ha sido el siguiente:

### = Diseño de esquemas.

En el menú de ventanas de Cadence Edge existe uno para la realización de esquemas. En el realizamos los esquemas de los módulos del diseño utilizando las librerías de los transistores de la tecnología Polyuse L de Thomsom que se encuentra situada en el directorio `\home\soft\TCM\polyu_v2.2\LIBTRANS`. Con respecto al proceso de diseño de los esquemas es conveniente mencionar ciertos problemas que han surgido y su solución.

En primer lugar comentar que en las propiedades de las resistencias debemos colocar su valor nominal y este debe indicarse con la letra en **minúscula** ya que si se escribe en mayúscula el simulador no les dara valor aunque si las reconozca. Por otra parte debemos tener cuidado a la hora de dibujar las líneas de unión de componentes ya que debe

comprobarse que se realicen correctamente y no queda ningún tramo abierto al igual que debemos tener en cuenta colocar nodos en los cruces de líneas que deben estar conectados. En general la realización de esquemas no tiene complicación excesiva y esta parte ha sido realizada sin dificultad.

-Realización de ficheros de entrada (*netlist*).

Para obtener estos ficheros debemos en primer lugar abrir el fichero del esquema del que queremos obtener el *netlist*. Una vez en él, se debe realizar una comprobación del esquema con la utilidad *extract* de la ventana de diseño de esquemas, la cual nos dará información de los problemas o peligros (*warnings*) que tiene el esquema para su corrección si se estima conveniente. A continuación en la ventana de simulación seleccionamos el simulador para el cual vamos a realizar el fichero de entrada, Spice en nuestro caso. El segundo paso será inicializar el entorno para la simulación en Spice (*inic. enviroment*), y por último la realización del *netlist*. Antes de comenzar con esto nos dará opción a cambiar el fichero del esquema, las librerías a utilizar y si lo queremos realizar en *Background* (con esto se refiere a realizarlo en "la parte trasera") para poder seguir realizando otras tareas.

Al terminar este proceso habremos obtenido el fichero llamado *netlist* dentro de el directorio *spice.run1* perteneciente al directorio del esquema. Este fichero lo podremos visualizar y modificar con cualquier editor de texto. La ruta de el fichero *netlist* en la estructura de la cuenta que ha sido utilizada en el laboratorio del CMA es el siguiente:

... \fmoreira\design\nombre diseño\schematic\spice.run1\netlist

Un ejemplo de *netlist* para simulación en Spice obtenido en el entorno Cadence

Edge es el siguiente.

### FICHERO DE ENTRADA DE UN AMPLIFICADOR DE BANDA ANCHA.

```
* net 0 = gnd!
* net 1 = /vcc
* net 2 = /vs1
* net 3 = /vs2
* net 4 = /ve1
* net 5 = /ve2
* net 6 = /Q1.E
* net 7 = /Q2.E
* net 8 = /Q34.E
* net 9 = /Q35.E
* net 10 = /Q0.C
* net 11 = /Q1.C
* net 12 = /R45.MINUS
* net 13 = /Q31.B
* net 14 = /Q0.E
* net 15 = /R25.MINUS
* net 16 = /R23.MINUS
* net 17 = /Q31.C
* net 18 = /R27.MINUS
* net 19 = /R28.MINUS
* net 20 = /Q31.E
* net 21 = /Q32.E
* net 22 = /R39.PLUS
* net 23 = /Q2.C
* net 24 = /Q3.C
* net 25 = /Q33.E
* net 26 = /R41.PLUS
```

```
* net 27 = /vee
* net 28 = /R12.MINUS
* net 29 = /R17.MINUS
* net 30 = /R15.MINUS
* net 31 = /R18.MINUS
* net 32 = /R16.MINUS
* net 33 = /R22.PLUS
* net 34 = /R21.PLUS
* net 35 = /R6.MINUS
* resistor(0) = /R44
R$#0 27 9 800
* resistor(1) = /R43
R$#1 27 8 800
* resistor(2) = /R42
R$#2 27 26 400
* resistor(3) = /R41
R$#3 26 25 200
* resistor(4) = /R20
R$#4 31 3 16k
* resistor(5) = /R19
R$#5 29 2 16k
* resistor(6) = /R18
R$#6 32 31 2k
* resistor(7) = /R17
R$#7 30 29 2k
* resistor(8) = /R16
R$#8 11 32 4k
```

```
* resistor(9) = /R15
R$#9 10 30 4k
* resistor(10) = /R37
R$#10 24 1 2k
* resistor(11) = /R36
R$#11 23 1 2k
* resistor(12) = /R40
R$#12 27 22 400
* resistor(13) = /R39
R$#13 22 21 800
* resistor(14) = /R38
R$#14 27 20 400
* resistor(15) = /R28
R$#15 17 19 800
* resistor(16) = /R30
R$#16 19 33 800
* resistor(17) = /R27
R$#17 17 18 800
* resistor(18) = /R29
R$#18 18 33 200
* resistor(19) = /R22
R$#19 33 6 100
* resistor(20) = /R25
R$#20 17 15 800
* resistor(21) = /R23
R$#21 17 16 800
* resistor(22) = /R26
```

R\$#22 15 34 200  
 \* resistor(23) = /R24  
 R\$#23 16 34 200  
 \* resistor(24) = /R21  
 R\$#24 34 14 100  
 \* resistor(25) = /R45  
 R\$#25 13 12 1k  
 \* resistor(26) = /R46  
 R\$#26 12 1 8k  
 \* resistor(27) = /R12  
 R\$#27 11 28 8k  
 \* resistor(28) = /R14  
 R\$#28 28 1 4k  
 \* resistor(29) = /R13  
 R\$#29 28 1 2k  
 \* resistor(30) = /R11  
 R\$#30 11 1 4k  
 \* resistor(31) = /R8  
 R\$#31 35 1 8k  
 \* resistor(32) = /R6  
 R\$#32 10 35 8k  
 \* resistor(33) = /R7  
 R\$#33 35 1 2k  
 \* resistor(34) = /R5  
 R\$#34 10 1 4k  
 \* qn2(35) = /Q5  
 Q35 1 24 3 N2  
 \* qn2(36) = /Q4  
 Q36 1 23 2 N2  
 \* qn1(37) = /Q35  
 Q37 3 13 9 N1  
 \* qn1(38) = /Q34  
 Q38 2 13 8 N1  
 \* qn1(39) = /Q33  
 Q39 7 13 25 N1  
 \* qn1(40) = /Q32  
 Q40 13 13 21 N1  
 \* qn1(41) = /Q31  
 Q41 17 13 20 N1

\* qn1(42) = /Q3  
 Q42 24 11 7 N1  
 \* qn1(43) = /Q2  
 Q43 23 10 7 N1  
 \* qn1(44) = /Q1  
 Q44 11 5 6 N1  
 \* qn1(45) = /Q0  
 Q45 10 4 14 N1

Observamos como para hacer comentarios se utiliza la anteposición de un asterisco. La primera parte del fichero son comentarios del nodo al que pertenece cada terminal de los componentes. Los transistores son descritos de tal forma que el primer nodo es el del colector, el segundo de la base y el tercero del emisor. Al final de cada transistor se especifica el tipo de transistor (N1 , N2 , ...).

Este fichero simplemente nos describe el circuito pero tal como está no estaría preparado para la simulación. Debemos para ello darle todas las señales de entrada, indicar las librerías a utilizar, la alimentación y los tipos de analisis que queremos realizar.

Este fichero

```
.Lib Tsfl.lib
vin1 vin1 0 ac .01 sin(0 .01 10e6)
vin2 vin2 0 ac .03 sin(0 .03 15e6)
Vcc vcc 0 7.5
Vee 0 Vee 7.5
```

puede servir como ejemplo de entrada de un netlist donde observamos las entradas de señal, la alimentacion a +/- 7.5

voltios y unos análisis transitorio y de alterna. Observa como las llamadas tarjetas de Spice van precedidas por un punto.

.- Simulación del circuito.

A partir del paso anterior el fichero está preparado para ser simulado. El proceso a seguir es el siguiente: una vez dentro de la ventana de simulación del entorno Cadence Edge elegimos la opción *run netlist* que es la que va a ejecutar la simulación. Previa

comprobación de los directorios a utilizar en la simulación el circuito comenzará a ser ejecutada. La respuesta final del simulador será la finalización satisfactoriamente ó no. A continuación describimos algunos de los casos que nos han producido que la simulación no se haya realizado adecuadamente y como lo hemos solventado:

- La falta de parametros en el fichero de entrada de un circuito es una de las causa mas comunes. En cualquier manual de Spice se debe consultar cual es la forma correcta de introducir todos los parametros.

- Caso muy especial es el de omitir en la definición de una señal la parte de "AC amplitud" y solo indicar que por ejemplo es una señal senoidal. Pues bien, en este caso la simulación se realiza correctamente, pero si pretendemos hacer un estudio en alterna la visualización de éste será errónea ya que si no introducimos en la definición de señal que se trata de AC el simulador realiza el analisis en alterna, tomando a esta señal como continua.

- "La opción LMTPTS demasiado pequeña". En circuitos de mayor magnitud puede darse este error. Para evitarlo pondremos una opción que aumente el lmtpts del circuito:

```
.OPTIONS LMTPTS=100000
```

- Quizás el error más complicado de solucionar a veces es el de "No hay convergencia en el análisis en continua". El análisis en continua del Spice en el entorno Cadence determina el punto de operación en continua del circuito con inductores cortocircuitados y capacitores abiertos. Una solución inicial del punto en continua se necesita antes que el análisis en alterna de pequeña señal para linealizar los modelos para dispositivos

no lineales. Así que la convergencia en continua es necesaria antes de realizar los análisis en alterna y transitorios. El programa contiene varias maneras para resolver los problemas de convergencia.

\* Una primera causa de la no convergencia es la de errores en el fichero del circuito. Conexiones incorrectas de dispositivos y elementos, el mismo nodo usado varias veces en el circuito son errores comunes. Los siguientes comandos pueden ayudarte a descubrir errores.

*Display nodes*: Imprime cada nodo seguido de la lista de elementos ó dispositivos a los cuales el nodo está conectado.

*Display element devices*: Muestra el nombre del elemento ó dispositivo seguido de las conexiones de nodos. El valor del elemento y el nombre del área y modelo son también dados.

*Display model*: Nombre del modelo , tipo y variables asociadas.

Después de comprobar que los problemas de convergencia no son un error de circuito puedes usar uno de los métodos siguientes.

**NOTA: Aconsejamos que al plantearse este problema la revisión de que el circuito este correcta sea minuciosa , ya que en la realización de este proyecto el simple cambio de la posición de un transistor , el colector**

donde estaria el emisor y viceversa, nos producía un error de convergencia que durante largo tiempo fue imposible resolver. Por lo tanto no se trata de mirar solo si las conexiones estan correctas, sino que tambien debemos comprobar la posición de los elementos.

\* Usando el comando *CONVRG*. Este hace que el proceso de convergencia sea mas simple.

\* Usando los comandos *STORE / RESTORE*. El *Store* graba los voltajes de los nodos en un fichero y con el *Restore* los recuperamos.

\* Usando el comando *NODSET*. Con el podemos forzar un valor de tensión en un nodo y en base a el se realizaria el analisis.

- El último problema digno de reseñar es el debido al tiempo de paso interno del analisis transitorio ( "*INTERNAL TIMESTEP TOO SMALL IN TRANSIENT ANALYSIS* "). Es un error muy común en los analisis transitorios. El tiempo de paso usado por el programa no es el mismo que nosotros le damos en el analisis a realizar (.tran 0.01seg 10 seg). El programa utiliza el que le damos (0.01 seg.) para la impresión en pantalla. El tiempo de paso interno es calculado dependiendo de la corriente de funcionamiento del circuito. Ampliando el valor del limite de puntos en el comando *.OPTIONS* lograremos solventar este problema.

Tras realizar la simulación en el directorio de simulación aparecen

estos ficheros y directorios que a continuación relacionamos y describimos:

- **Netlist.** Es el fichero que nos describe el circuito. Su traducción literal sería lista de nodos.
- **si.env.** Es el fichero que se va a simular en el esta los datos del simulador a utilizar, las librerías, el fichero de entrada,...
- **si.inp.** Fichero de entrada de simulación. En el observaremos los modelos de los dispositivos a utilizar, la descripción del circuito, las entradas del circuito y el límite de puntos establecidos.
- **si.log.** En este fichero quedan guardados todos los mensajes que nos manda el simulador durante la simulación (tiempo de simulación, simulación correcta ó no,...).
- **si.out.** Es el fichero de salida. En el tenemos todos los análisis realizados, los datos del estudio en continua, el número de elementos del circuito, los *.plot* de las señales que se han pedido, potencia,...
- **spice.inp.** Es un fichero de entrada. Para no estar modificando el *netlist* cada vez que vayamos a cambiar las entradas ó cambiar las opciones, utilizaremos este fichero.
- **map.** Es un directorio en el cual quedará incluido el fichero *waves* donde estarán los datos para el dibujo de las formas de onda.

Cuando modificamos un fichero se crea otro de igual nombre pero con el añadido % donde estará el fichero sin las modificaciones. Por ejemplo *Netlist-Netlist%*.

-Visualización de resultados.

El Cadence Edge posee una utilidad para ver formas de onda llamada " *WaveForm Display* ". Tras la simulación abrimos la ventana de visualización, pero no en la misma pantalla en que estamos sino que abrimos una nueva y en ella inicializamos el entorno para la visualización. De otra forma el entorno da problemas para intentar visualizar resultados. Una vez inicializado podemos visualizar la señal en todos los puntos siempre que haya un punto marcado en el circuito.

Debido a que este visualizador no tiene la propiedad de obtener las respuestas en frecuencia nos vemos obligado a cambiar de simulador (HSPiCE) y en consecuencia usar un visualizador de ondas de mayor potencia (GSI).

A pesar de esta limitación nos sirve para comprobar las formas de onda asimétricas y ayudados de los comandos *.print* y *.plot* ver las señales diferenciales en el fichero de salida, lo cual serviría para verificar el correcto funcionamiento del circuito.

### **4.3.- Simulaciones en Hspice.**

El simulador elegido para reemplazar al Spice del sistema Cadence es el Hspice. Este *software* lo ejecutaremos desde fuera del Cadence debido a que su utilización desde dentro del sistema, supone una excesiva ocupación de memoria por parte de los ficheros de formas de onda, e incompatibilidad con las librerías suministradas por la casa Thomsom.

Para ejecutar el Hspice necesitamos un fichero de entrada. La obtención de este fichero de entrada es el primero de los inconvenientes a evitar, ya que hasta ahora lo realizaba automáticamente el sistema Cadence. La solución obtenida es que debemos introducir el esquema en Cadence, y realizar un *netlist* para Spice. Este netlist es el que tras modificaciones oportunas se utilizará paara la simulación de Hspice desde fuera del Cadence.

### EJEMPLO DE FICHERO DE ENTRADA PARA HSPICE

```
.options post
.include libreria
* net 0 = gnd!
* net 1 = /vcc
* net 2 = /vin
* net 3 = /vee
* net 4 = /vout
* net 5 = /Q0.E
* net 6 = /Q1.E
* net 7 = /Q3.E
* net 8 = /Q5.E
* net 9 = /Q2.B
* net 10 = /Q0.C
* resistor(0) = /R10
R$#0 10 1 5k
* resistor(1) = /R8
R$#1 3 5 750
* resistor(2) = /R7
R$#2 6 3 100
* resistor(3) = /R6
R$#3 3 8 1k
* resistor(4) = /R9
R$#4 9 1 7k
* qn1(5) = /Q5
Q5 7 10 8 N1
* qn1(6) = /Q4
Q6 1 2 7 N1
* qn1(7) = /Q3
Q7 9 4 7 N1
* qn5(8) = /Q2
Q8 1 9 4 N5
* qn5(9) = /Q1
Q9 4 10 6 N5
* qn5(10) = /Q0
Q10 10 10 5 N5
*.tran .001u 1u oct 10 1 1e9
vcc 1 0 7.5
vee 0 3 7.5
vout 4 0 ac 30m
vin 2 0 0
*vin 2 0 ac 30m sin(.0 30M 3ME)
*.plot tran v(4)
.end
```

Tras estudiar cual debe ser el formato de un fichero de entrada para Hspice, observamos que son bastante parecidos excepto en las modificaciones que comentamos a continuación:

.-Es obligatorio incluir el comando *.end* al final del fichero de entrada.

.- Si queremos visualizar las formas de onda tras la simulación debemos incluir con el comando *.options* la opción *post*.

.- Ahora necesitamos indicarle donde estan las librerías de modelos a utilizar.

Para ello hemos localizado el fichero, suministrado por la tecnología donde nos describe cada modelo de dispositivo (*/home/soft/tcm/polyu\_v2.2/LIBTRANS* es la situación de este fichero en nuestro sistema). Comparando este con otros ficheros de modelos para hspice vemos que pueden ser igualmente válidos. Para que la simulación tenga en cuenta estos modelos debemos incluir este fichero utilizando el comando *.include*.

Tras estas modificaciones un fichero de entrada para simular en Hspice puede tener la forma indicada en el recuadro anterior.

Segun los análisis que se hagan en la simulación, tras esta podremos observar que se han creado ficheros *.tr0* para los analisis transitorios, ficheros *.ac0* para los análisis en alterna, ficheros *.sw0* para análisis en continua, y asi distintas extensiones segun el análisis realizado.

Para medir la potencia en simulaciones Hspice se ha utilizado el analisis *.meas tran.....*

El Hspice suministra el visualizador de formas de onda GSI. Este visualizador es capaz de visualizar señales diferenciales principal limitación del *Waveform Display*. Para ejecutar *GSI* basta con introducir *gsi* mas el nombre del fichero a visualizar, ó simplemente *gsi*.

En el entorno de ventanas del *gsi* podemos ver una ventana para moverse a traves de los distintos directorios y subdirectorios hasta encontrar el fichero deseado, una ventana de control donde marcaremos las señales a visualizar y las ventanas de visualización.

Desde el *gsi* se puede ejecutar el hspice indicandole cual es el fichero a simular y cual el de salida. De esta manera no tenemos por qué salir del *gsi* para ejecutar una nueva simulación en hspice. Al contrario que en el Wavefrom Display del Cadence no tenemos por qué marcar los nodos de las señales a visualizar, sino que elegimos los deseados desde el cuadro de diálogo del *gsi*.

En el cuadro de diálogo para la selección de nodo tiene dividido un cuadro para señales asimétricas, otro para voltajes, otro para corrientes, otro para señales independienes, otro que incluye todo, quedando un último para expresiones del tipo de los decibelios o las señales diferenciales.

#### **4.4.- Realización del trazado físico.**

Una vez simulado el diseño comenzaremos con la realización física del

circuito. Para ello desplegamos el menú *layout* y editamos el diseño obteniendo en pantalla el *layout* L1200 suministrado por Thomsom con la distribución de todos los componentes en *tiles*. Nuestro trabajo consistirá en realizar la metalización de las conexiones entre los componentes.

La interconexión de los componentes corresponde a la segunda capa de metalización. La primera capa de metal, ya programada por TMS, proporciona el uso de cruces en el ruteado y buses de alimentación.

Una rejilla ha sido dibujada sobre el *layout* para simplificar las operaciones sobre él. Así que cada línea del canal representa un canal válido para interconectar componentes.

Sobre la base de esta rejilla, los usuarios pueden interconectar dispositivos para el dibujo de líneas sobre el *layout* para posterior digitalización o para usar un editor de gráficos.

#### - Reglas del Layout.

La lámina de ruteado esta hecha de un modelo cuadrado de rejilla superpuesto, sobre el esquema simbólico, de este modo indican el paso nominal de interconexiones.

Los anchos actuales de conexiones estan definidos por defecto sobre las herramientas finales de CAD sin intervención del diseñador. Si es necesario dos cuadritos

adyacentes del grid pueden ser usados para ruteado de alta corriente. La mayor parte de la superficie puede también ser implementada.

Las interconexiones deseadas deben ser dibujadas sobre a lámina de símbolos del layout , indicando claramente la localización del patillaje (ó en na transparencia sobre esa lámina).

El diseñador esta asistido mayormente por las características disponibles sobre la primera capa de metalización.

Los cruces pueden ser usados en caso de inevitables dificultades de ruteado.

Cada transistor NPN tiene varios puntos de acceso a su colector. Igualmente accesos a la base de QPL tipo PNP de transistor esta disponible a través de varios puntos. Cuando el transistor es usado, estos puntos de acceso pueden ser elegidos indistintamente. Cuando el transistor no es usado , estos puntos pueden ser utilizados como canales de ruteado adicionales.

Como siempre se debería tener en cuenta que estos canales contienen capacidades parásitas equivalente a la capacidad colector - substrato par los NPN y base - substrato en el caso de los PNP.

La alimentación está distribuida por cuatro buses diferentes:

-Alimentación Negativa:  $V^-$  (Substrato)

$V^-$  máxima tal que

$$V^+ - V^- < 15 \text{ V.}$$

- Alimentación Positiva:  $V_1^+$  max. 15 V

$V_2^+$  max. 15 V.

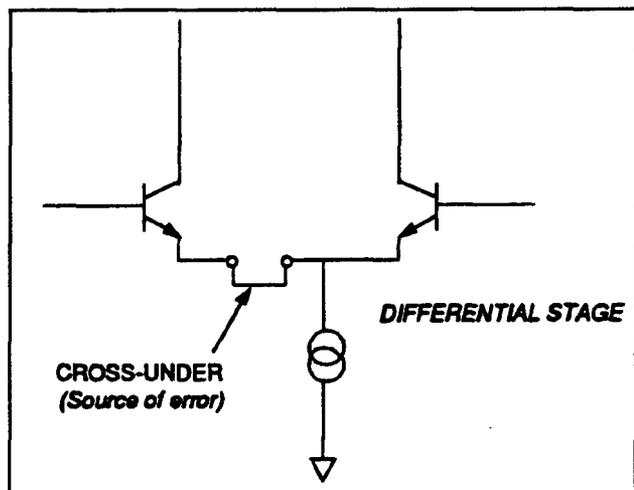
$V_3^+$  max. 15 V.

- Recomendaciones para el *layout*.

El *layout* deberá seguir las líneas punteadas sobre el grid. En particular las líneas no deberán estar dibujadas entre dos puntos.

No deben aparecer pistas en el área gris alrededor de los pads, reservados para conexiones. La única excepción es cuando la unión debe estar conectada a el pad central.

Dos interconexiones, no deben estar bajo ninguna circunstancia localizadas sobre el mismo punto.



Las interconexiones no deberán coincidir con pads ó alimentación, a menos que este conectada a uno de ellos.

El diseñador es aconsejado a dividir la totalidad del diseño en bloques funcionales y colocar estos bloques sobre la lámina de ruteado, mientras, prestando atención a la distribución de potencia de alimentación y la localización de los pads de entrada y salida.

Los bloques deben ser localizados tal que las interconexiones entre ellos sean fácilmente realizables.

Se debe prestar especial atención a las resistencias adicionales introducidas por cruces en el ruteado (*cross-unders*). Los siguientes ejemplos ilustran dos casos donde el *offset* es introducido por cruces.

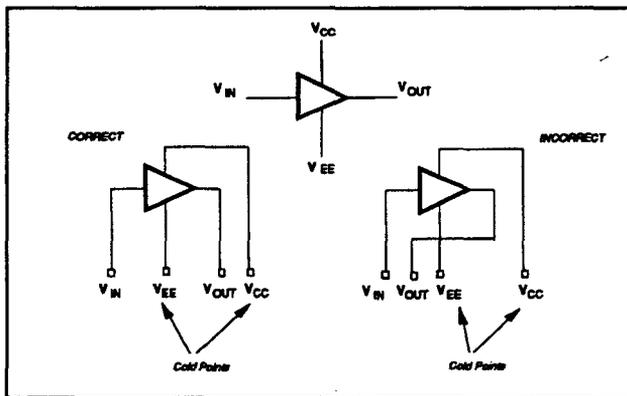
A veces la conexión de resistencias no es siempre insignificante, ya que capacidades parásitas pueden aparecer a veces entre componentes. Esto es particularmente cierto para conexiones de masa y alimentación.

Además, si componentes de baja resistencia son usados como *cross-unders* para simplificar el ruteado, es aconsejable usar el simulador para comprobar que las propiedades del diseño no son modificadas por este procedimiento.

Si uno de los diferentes componentes disipan calor significativo mientras otros componentes requieren buenas características de emparejamiento, los componentes deben dividirse en dos grupos y entonces los componentes que deben estar pareados deben colocarse

a igual distancia de las fuentes de calor.

En alta frecuencia y donde los niveles de entrada y salida son bastante diferentes es aconsejable separar estas señales por "puntos frios" de tal forma que evite *cross-coupling* y así elimina cualquier riesgo de oscilación.



En el ejemplo se muestra como esta solución es aplicada a un amplificador de alta ganancia.

A la hora de realizar el ruteado del *layout* del circuito, el diseñador deberá tomar en consideración todos los componentes parásitos capaces de modificar las propiedades del circuito.

En efecto, aunque los componentes del array están dibujados para estar libres de efectos parásitos bajo condiciones normales de operación, deberá observar que efectos produce ya que resistencias y capacidades asociadas a conexiones, están siempre presentes.

#### 4.5.-Comprobación del diseño físico.

Para comprobar la validez del diseño físico realizado utilizaremos el menú PDV (*Physical Design Veridication*). Al desplegarlo realizaremos dos comprobaciones

fundamentales:

La primera es una comprobación de que se han cumplido las reglas de diseño. Para ello utilizaremos el DRC (Comprobación de Reglas de Diseño) con el fichero de reglas suministrado por el fabricante. Este fichero es el *polyu104.drc* localizado en el *path* */home/soft/tcm/polyuse\_v2.2/DR CERCLVS*. Si existe algún fallo se resaltarán y habrá que solucionarlo.

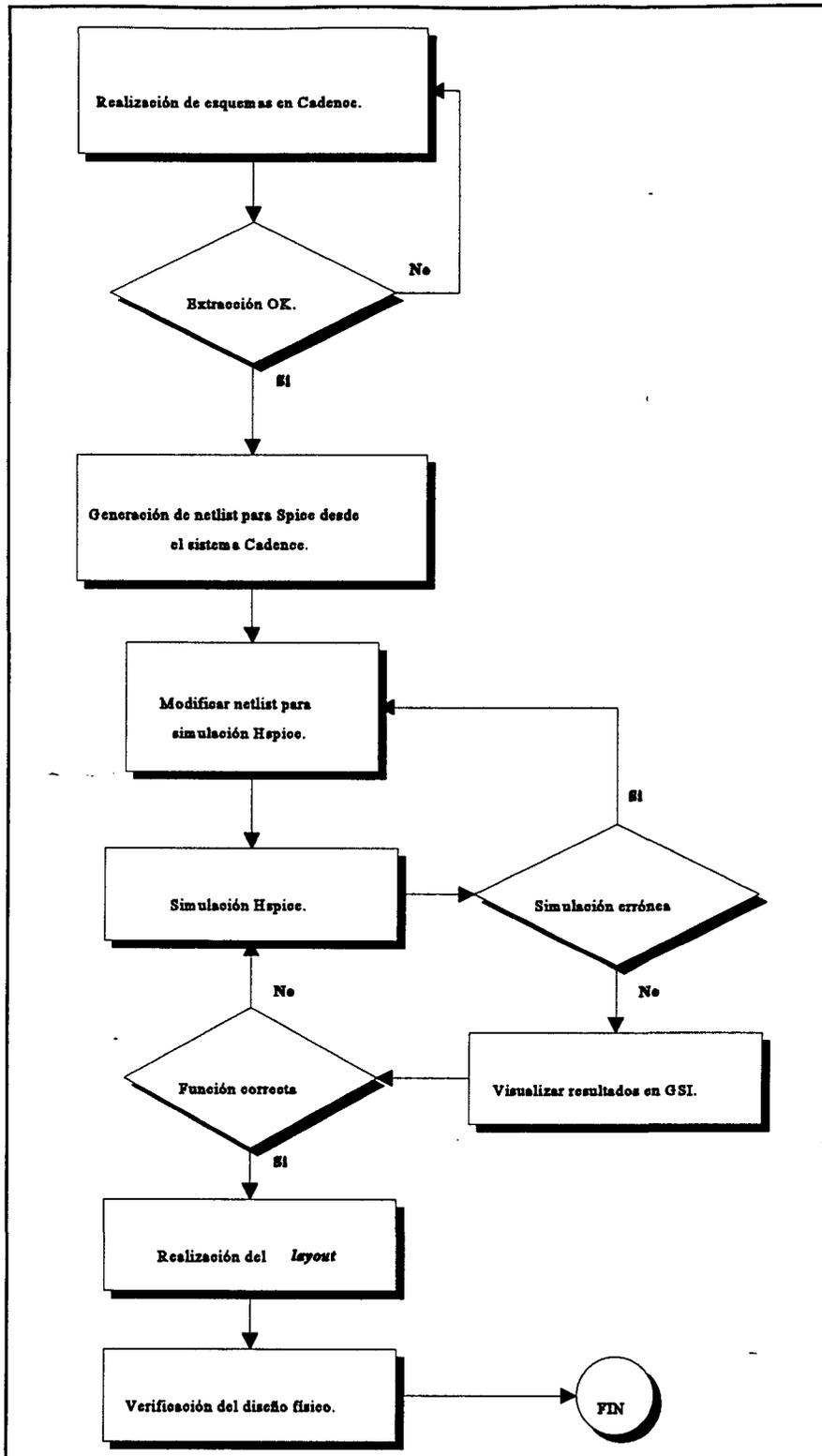
La segunda es una extracción del diseño del *layout*. Utilizaremos para ello el fichero *xtr* del mismo directorio que el anterior. Al realizar el *extracted* genera un fichero con las propiedades eléctricas y teniendo en consideración los elementos parásitos que puedan existir.

Una tercera comprobación es el Pdcompare que crea un fichero de entrada para el *extracted* y lo compara con el fichero de entrada del esquema, viendo así si son iguales.

## **5.6.- Elección del encapsulado.**

Para finalizar debemos buscar ubicación a nuestro diseño. Para ello existen diferentes tipos de encapsulados según las características que deseemos, y claro está, el número de patillas que posee. En uno de los anexos describimos los diferentes tipos de encapsulados de los que se disponen para ubicar el diseño.

El siguiente diagrama de flujo nos muestra la ruta de diseño seguida.



# Capítulo V

## Diseño de Macro células

### Basadas en la Tecnología Bipolar TSFL

En esta parte del proyecto vamos a diseñar y analizar una serie de células basadas en esta tecnología, de las cuales comentaremos sus posibles utilidades, prestaciones, márgenes de funcionamiento, etc...

#### 5.1.- Buffer de ganancia 1.

Esta célula se ha diseñado dada la necesidad de tener una etapa que sin modificar la ganancia de la señal tuviera alta impedancia de entrada y baja impedancia de salida para así poder atacar a etapas posteriores sin que esta cargue a su predecesora.

Sabiendo que a un amplificador operacional al que cortocircuitamos la entrada con la salida nos iba a dar estas prestaciones, el trabajo se basaba en tratar de conseguir un seguidor utilizando un operacional.

Para ello hemos utilizado la configuración básica de lo que es un operacional que simplemente es un amplificador diferencial.

Utilizando como fuente de corriente un transistor (espejo de corriente) y llevando la salida de este a una configuración de seguidor de emisor que nos va a servir para

que haya bastante corriente a la salida, y de paso establecer la referencia de tensión que va a haber en la fuente de corriente de amplificador diferencial.

Por último solo nos queda llevar una realimentación de la salida hasta la entrada quedando diseñado el seguidor.

Utilizaremos para esta célula el transistor denominado QN4 perteneciente a

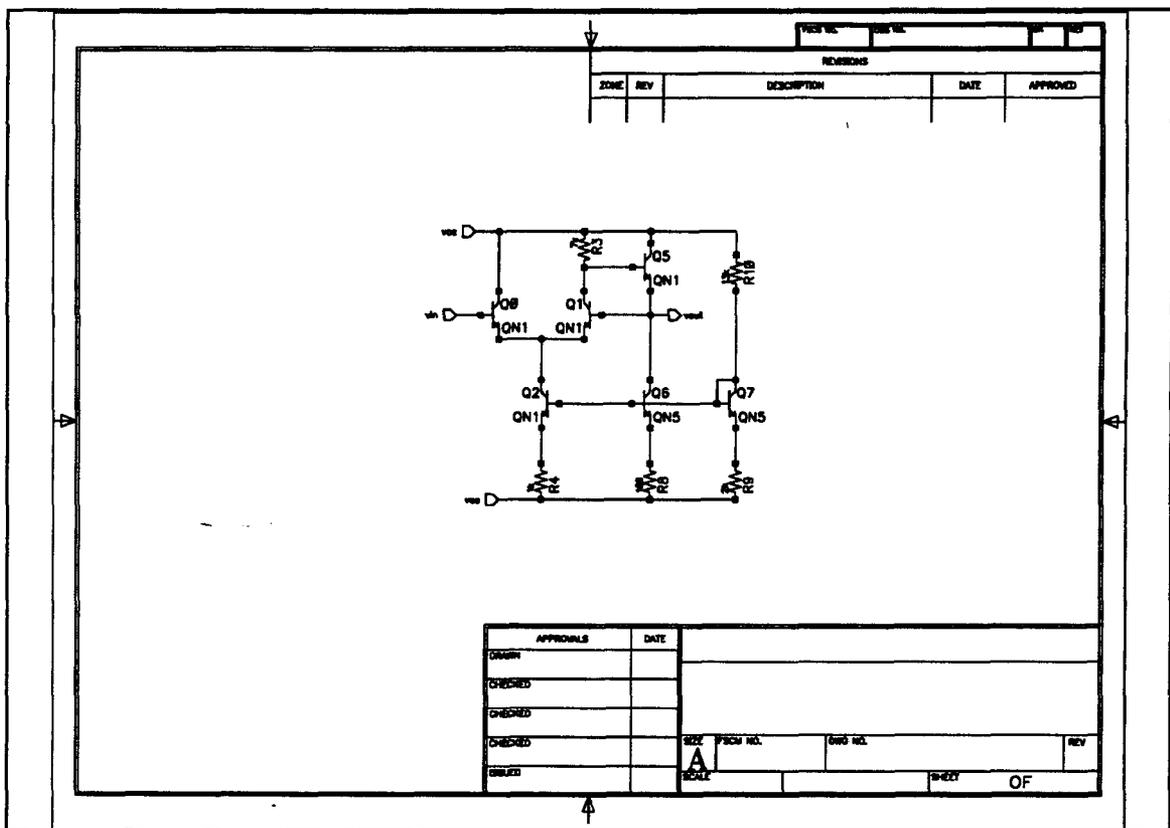


Figura 5.1.1.- Esquema del Buffer.

esta tecnología que tiene como característica más importante su mayor intensidad de colector,  $I_{c_{max}}=40 \text{ mA}$ .

A continuación mostramos los esquemas sucesivos y el proceso de cálculo para lograr que no haya nivel de continua.

El esquema del buffer diseñado es de la forma indicada en la figura 5.1.1.

© Del documento, los autores. Digitalización realizada por ULPGC. Biblioteca Universitaria, 2008

Lo primero que observamos en él es el par diferencial con una fuente de corriente en el emisor formada por un base común. En la base del primer transistor (Q0) será donde apliquemos la señal, y en el colector del Q1 donde tomaremos la salida del par diferencial.

Las resistencias R9, R10 y el transistor Q7 los utilizamos para fijar una corriente. Esto lo logramos cortocircuitando el colector y la base de Q7 de tal forma que la diferencia de potencial entre colector y emisor sea constante e igual a 0.8 voltios aproximadamente. De esta manera al tener las bases de los transistores Q6 y Q2 unidas al Q7 el potencial en la base de los tres transistores será la misma.

$$I = (V_{cc} - V_{ee} - V_{be}) / (R_9 + R_{10})$$

$$V_{base} = 0.8 + I * R_9 - V_{ee}$$

El transistor Q5 está configurado de tal manera que funciona como un seguidor de emisor. Toma la salida del par diferencial y sin ganancia de tensión ( $A_v=1$ ) obtenemos mayor intensidad a la salida (emisor de Q5) para poder atacar a etapas posteriores sin dificultad. Utilizamos una realimentación de tensión a través de un transistor (Q6) que funciona como fuente de corriente. De esta manera estabilizamos el circuito consiguiendo unas respuestas muy satisfactorias.

A la hora del diseño hemos de tener en cuenta que la Voffset a la salida sea mínima. Para ello tenemos varias formas de realizar el diseño dependiendo de la prioridad que se de a los problemas que pueden surgir. Si no se desea que el consumo de potencia suba, la solución se encontraría aumentando la R3 de tal manera que en el colector del par diferencial

solo quede la tension Vbe. Con esta solución aparecería el problema de que las resistencias en el *layout* son limitadas. Si no nos importa que el consumo de potencia aumente la solución estaría en aumentar la corriente que circula por la fuente. Todo dependerá de nuestros intereses.

Cabe la posibilidad de que nos interese que la Voffset tenga un cierto nivel para ajustar un nivel de continua indeseable de la etapa anterior.

De las células diseñadas es la que menos área ocupa, utilizando solo parte de un tile, estimando su area en 450\*450 micras.

La potencia de este diseño es de 306 mW. Este valor es debido a la utilización en el diseño de transistores de mayor potencia , como son los QN5.

A continuación exponemos una configuración con la que hemos obtenido un mínimo de *Voffset* llegando a un equilibrio entre consumo de potencia y la limitación de

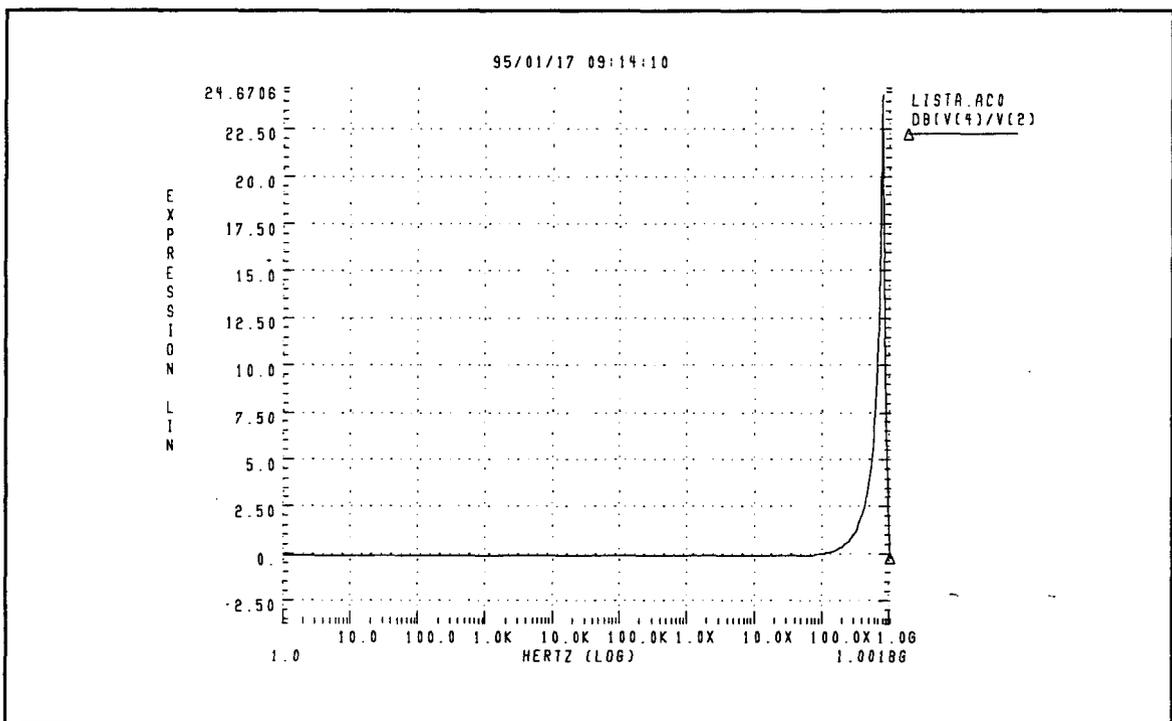


Figura 5.1.2.- Respuesta en Frecuencia.

espacio de los *layout*. Tambien incluimos las formas de onda de los distintos análisis realizados sobre esta célula.

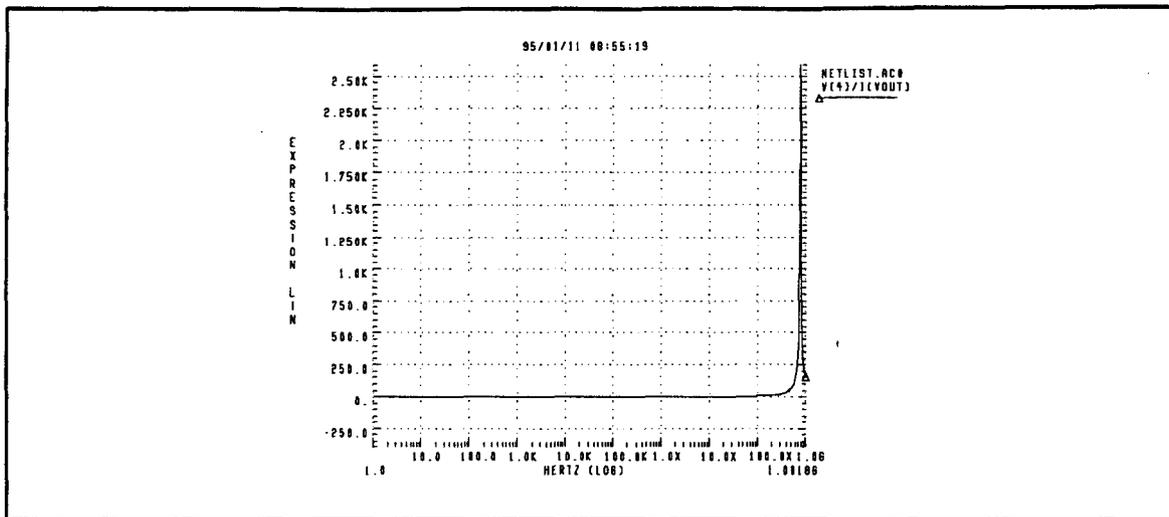


Figura 5.1.3.- Impedancia de Salida.

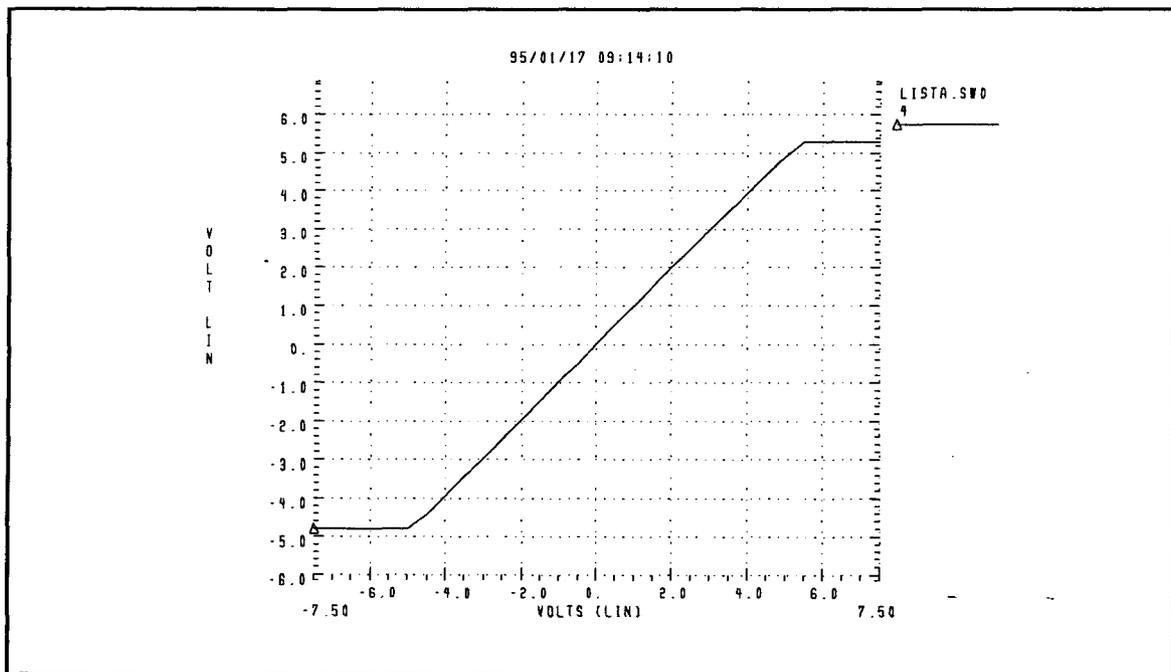


Figura 5.1.5.- Analisis en Continua.

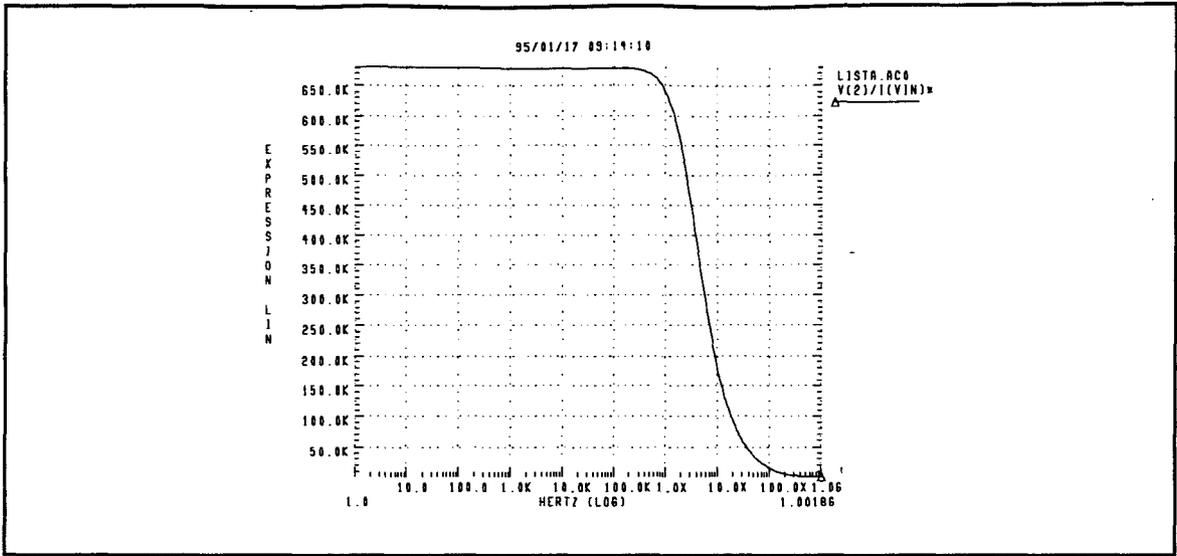


Figura 5.1.4.- Impedancia de Entrada.

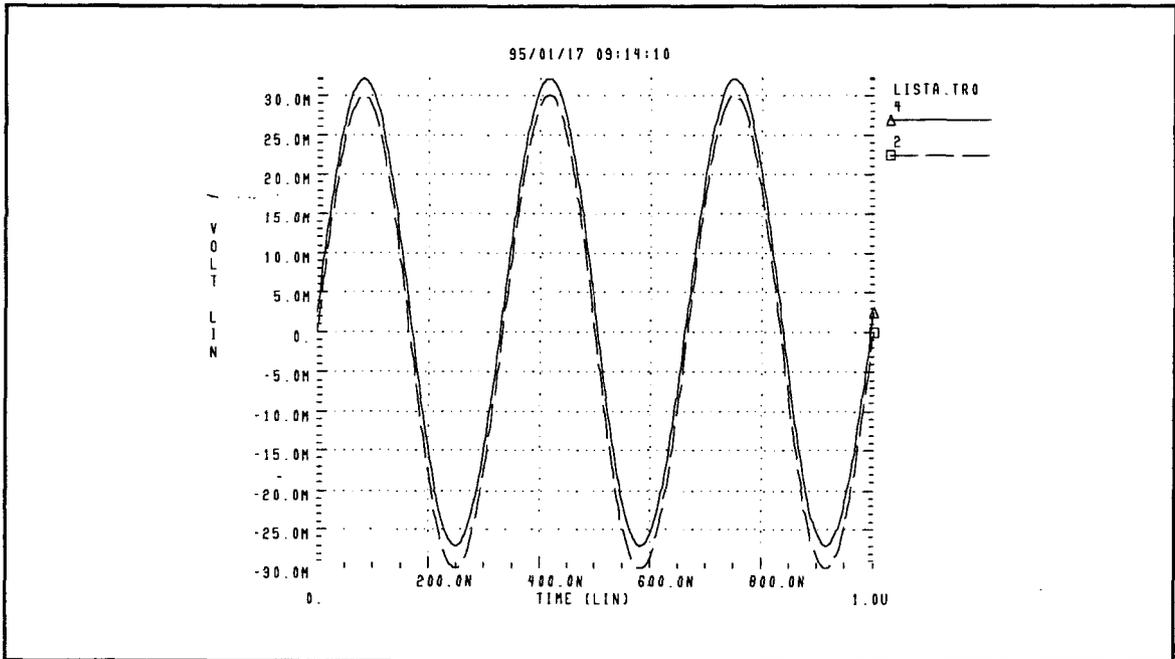
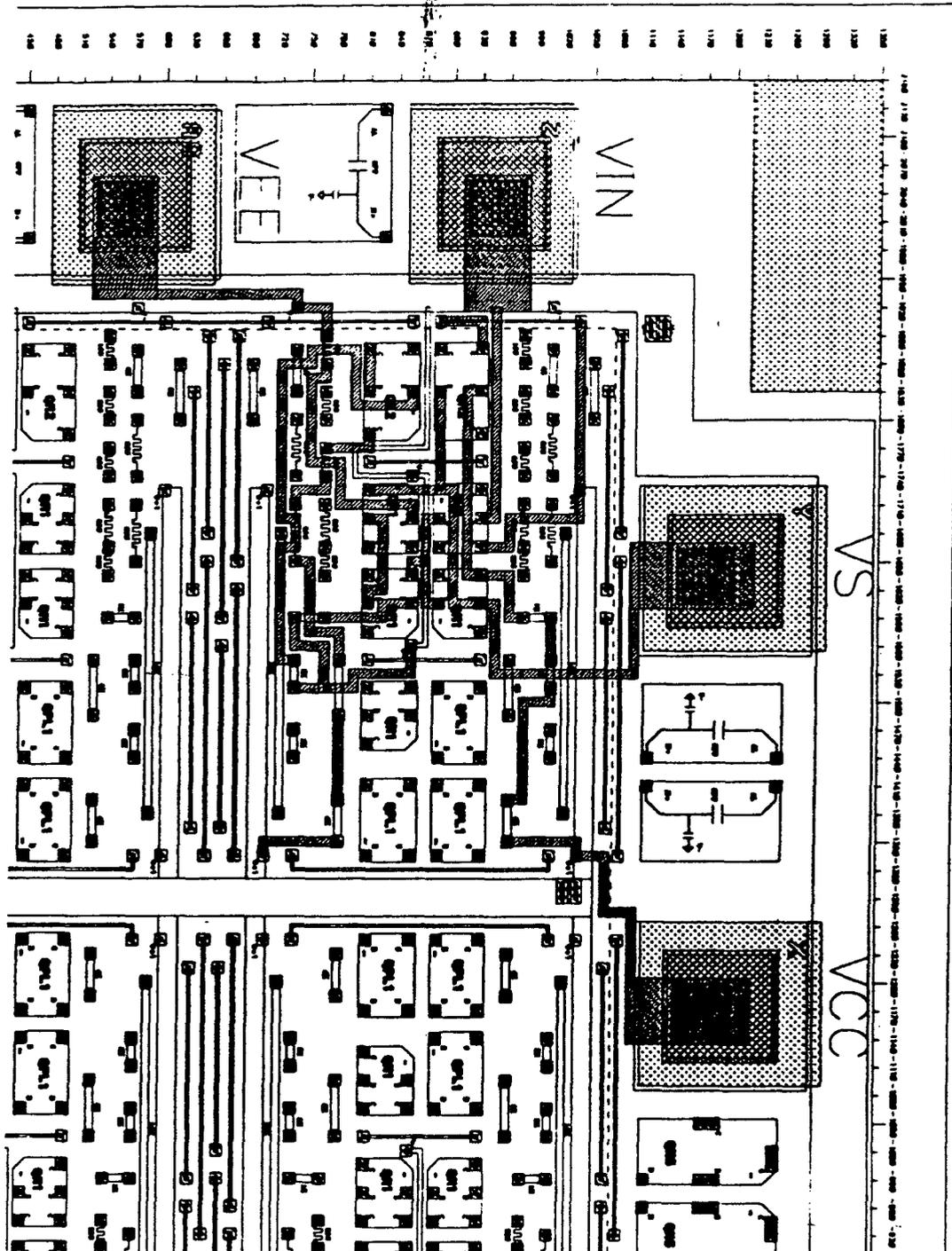


Figura 5.1.6.- Análisis Transitorio.



## Características del Buffer

	<b>Buffer</b>
<b>Nº de Transistores</b>	6
<b>Nº de Resistencias</b>	9
<b>Área Utilizada</b>	1 Tile
<b>Alimentación</b>	+7,5 / -7,5 Voltios
<b>Ganancia en dB</b>	1 dB
<b>Offset</b>	2,2 mV
<b>Ancho de Banda</b>	460 MHz
<b>Impedancia de Entrada (<math>Z_i</math>)</b>	680 Kohmios
<b>Impedancia de Salida (<math>Z_o</math>)</b>	Nula
<b>Potencia</b>	306 mW

Tecnología: Bipolar Polyuse L de Thomson

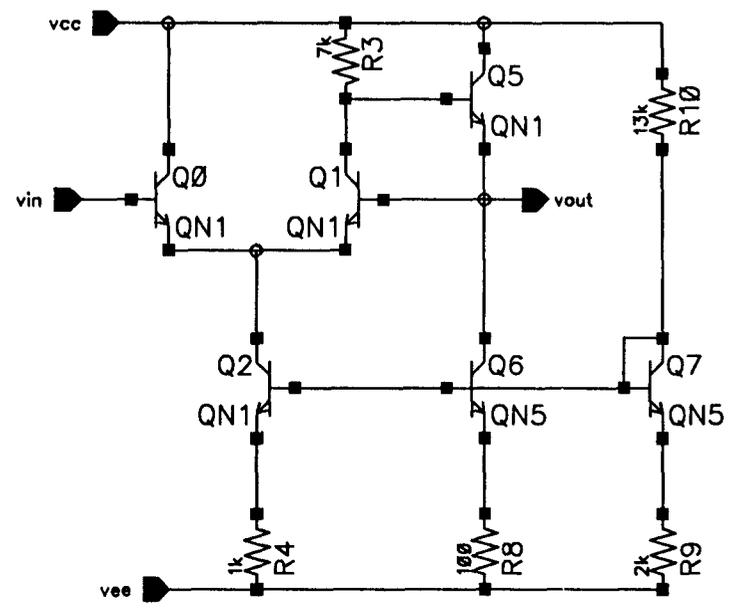
Simulaciones: Hspice

Fecha: Noviembre 1994

Observaciones: Ninguna

FSCM NO.	DWG NO.	SH	REV
----------	---------	----	-----

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPROVED



APPROVALS	DATE	<h1>BUFFER</h1> <h2>AV=1</h2>			
DRAWN					
CHECKED					
CHECKED					
CHECKED					
ISSUED		SIZE <b>A</b>	FSCM NO.	DWG NO.	REV
		SCALE	SHEET		OF

## 5.1.- Amplificador de Video.

El amplificador de video (de banda ancha) tiene una respuesta plana sobre un intervalo muy extenso de frecuencias. Los anchos de banda comunes estan dentro de la región de los Mhz.

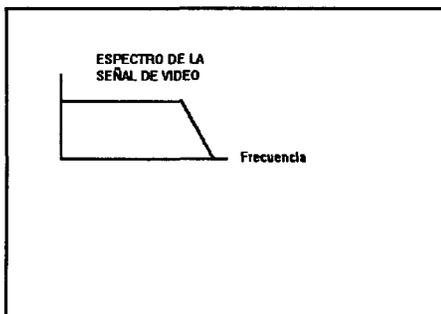


Figura 5.1 Señal video banda base.

Los amplificadores de video son utilizados en aplicaciones en las que el intervalo de frecuencias de entrada es muy grande. Por ejemplo, los osciloscopios operan de 0 a mas de 10 Mhz; estos instrumentos utilizan amplificadores de video para incrementar la

intensidad de la señal antes de enviarla al tubo de rayos catódicos (TRC). Otro ejemplo es el que nos va a ocupar, el cual consiste en utilizar el amplificador de video para manejar señales de alrededor de 4 Mhz correspondientes a la señal de video.

En nuestro caso la célula va a desempeñar funciones de amplificador para señales de alta frecuencia. Se sabe que estas señales son de frecuencia del orden de los Mhz por lo que la célula deberá ser lineal hasta estas frecuencias por lo menos. Si además se quiere que esta célula pueda tratar la señal de video a frecuencias intermedias el amplificador deberá ser lineal hasta los 30-40 Mhz como mínimo.

En la figura 5.1 se observa el espectro de la señal de video en banda base.

Cuando se transporta esta señal a frecuencia mas altas para su mejor transmisión ( frecuencias intermedias) el espectro queda situado a unos 30-40 Mhz segun el canal en que

estemos.

El diseño estará basado en dos amplificadores diferenciales en cascada, de los que la señal de salida será llevada a unos seguidores de emisor para dar corriente a la salida. De los seguidores hacia las bases de los transistores del segundo diferencial habrá una realimentación para evitar variaciones en la salida.

Se incluye en nuestro diseño la posibilidad de controlar la ganancia incluyendo en los emisores del primer par diferencial unas resistencias con la posibilidad de cortocircuitarlas según se quiera más o menos ganancia. Las corrientes de las fuentes las fijamos mediante un espejo de corriente.

Con la siguiente configuración se logra un ancho de banda del orden de 400 Mhz con lo cual será más que suficiente para la utilidad que se ha diseñado. Si se quiere que la ganancia sea mayor se podrá aumentar en detrimento del ancho de banda, ya que sabemos que el ancho de banda y la ganancia en un amplificador tienen una relación inversamente proporcional.

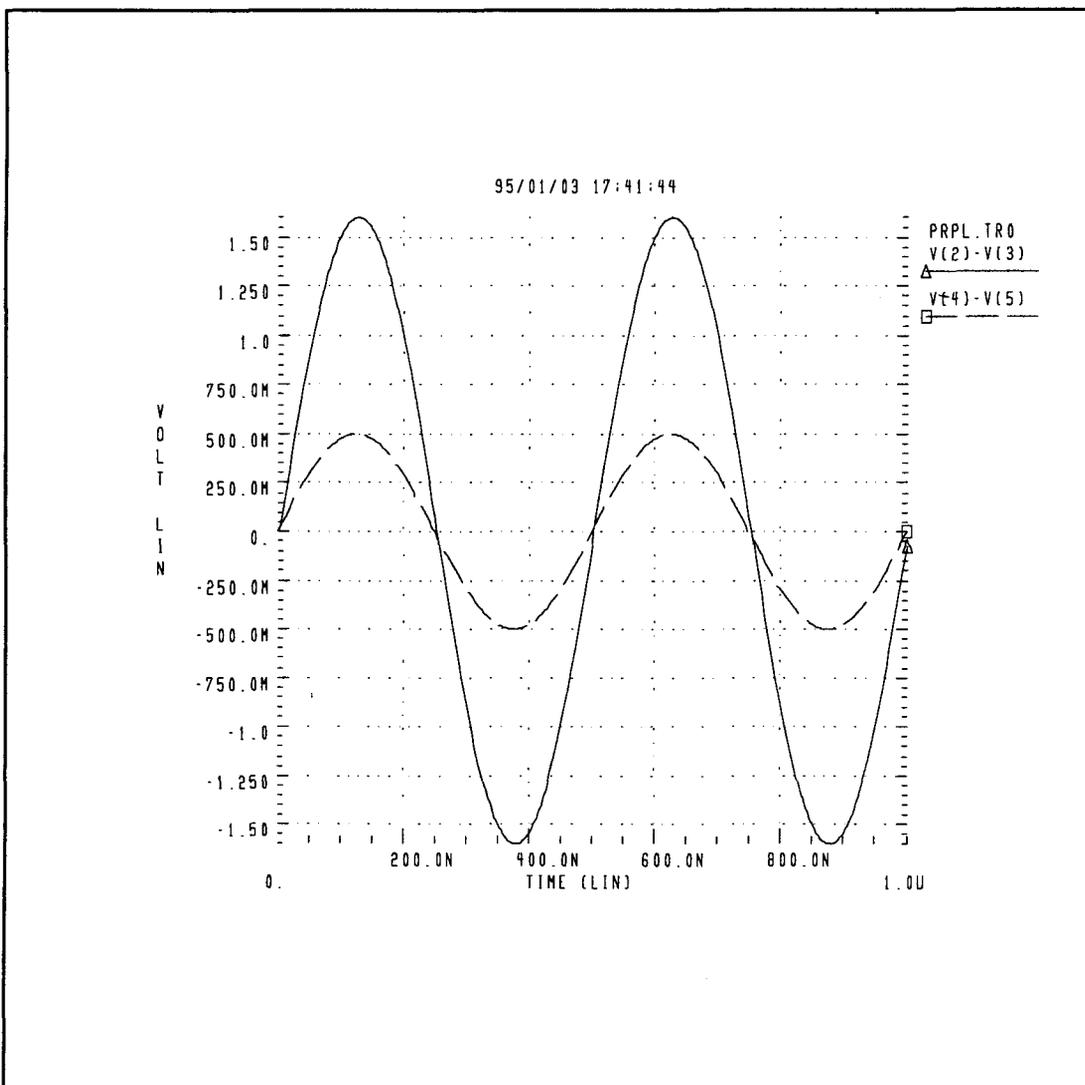
El nivel de offset a la salida es de 0 voltios. Si se desea algún nivel de continua a la salida deberíamos usar un desplazador que nos ajustara ese nivel. Cabe la posibilidad de que se necesite mayor corriente a la salida por ser esta etapa la final del diseño. Si esto ocurre se puede conectar el buffer anteriormente diseñado, el cual sin modificar la amplitud de señal hace que a la salida haya mayor corriente y así podamos conectar a etapas exteriores sin que carguen al DVA.

La impedancia de entrada para la configuración que se indica es de 100k. La impedancia de salida es de 50 ohmios. Al no ser la impedancia de entrada muy grande se tiene la posibilidad de incorporarle un buffer a la entrada para no tener problemas de adaptación

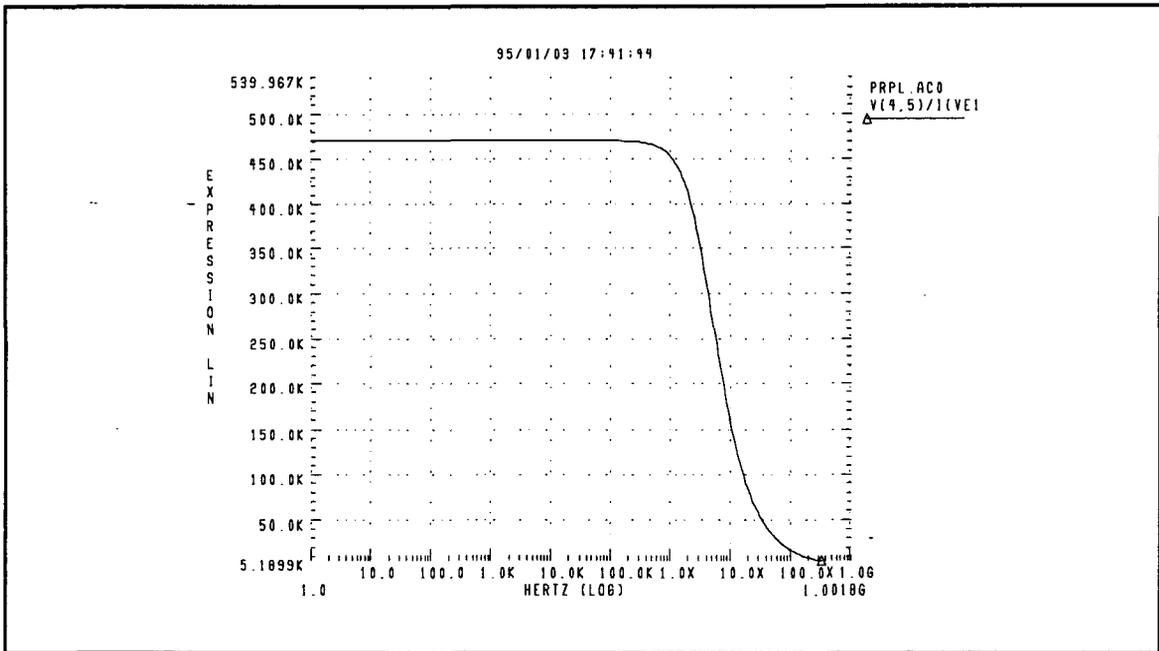
cuando lo incluyamos en cualquier circuito.

Cualquier variación que se haga en el circuito debe ser de tal forma que se tenga en cuenta el punto de continua donde va a trabajar, pudiendo dar problemas de saturación a la salida y un exceso de *offset* indeseable.

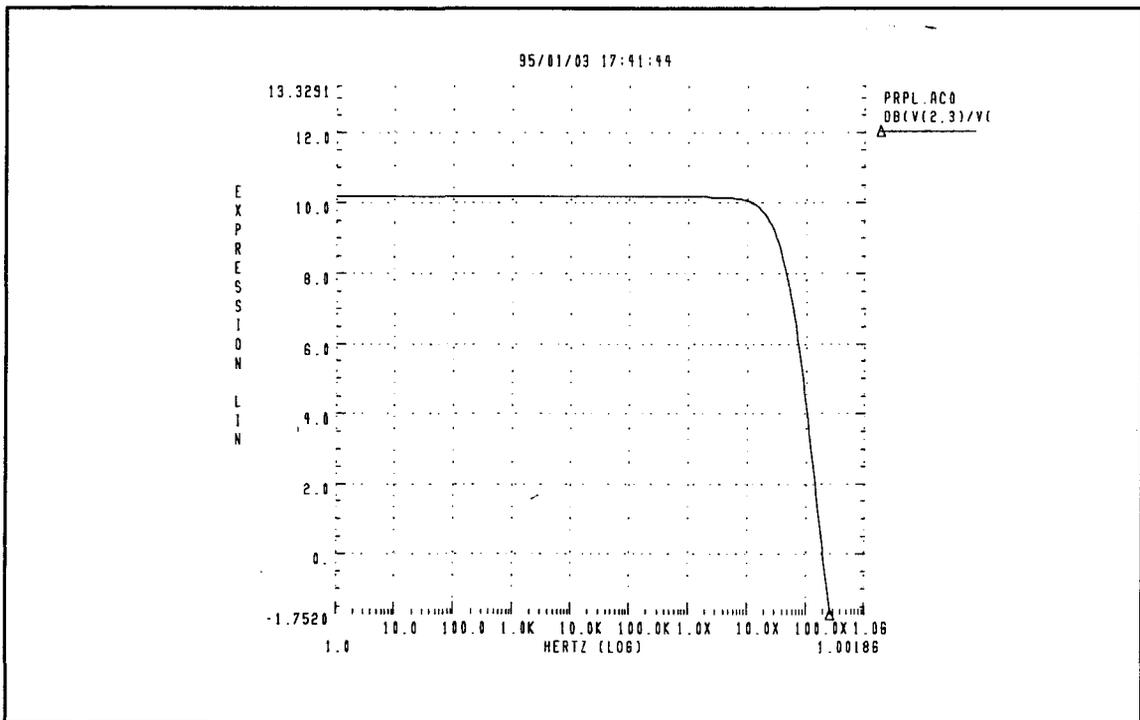
A continuación se muestra las respuestas en frecuencia y ganancia de esta etapa para diferentes configuraciones y para valores críticos de los parámetros de los transistores. Además se incluye las impedancias de entrada y salida de este circuito.



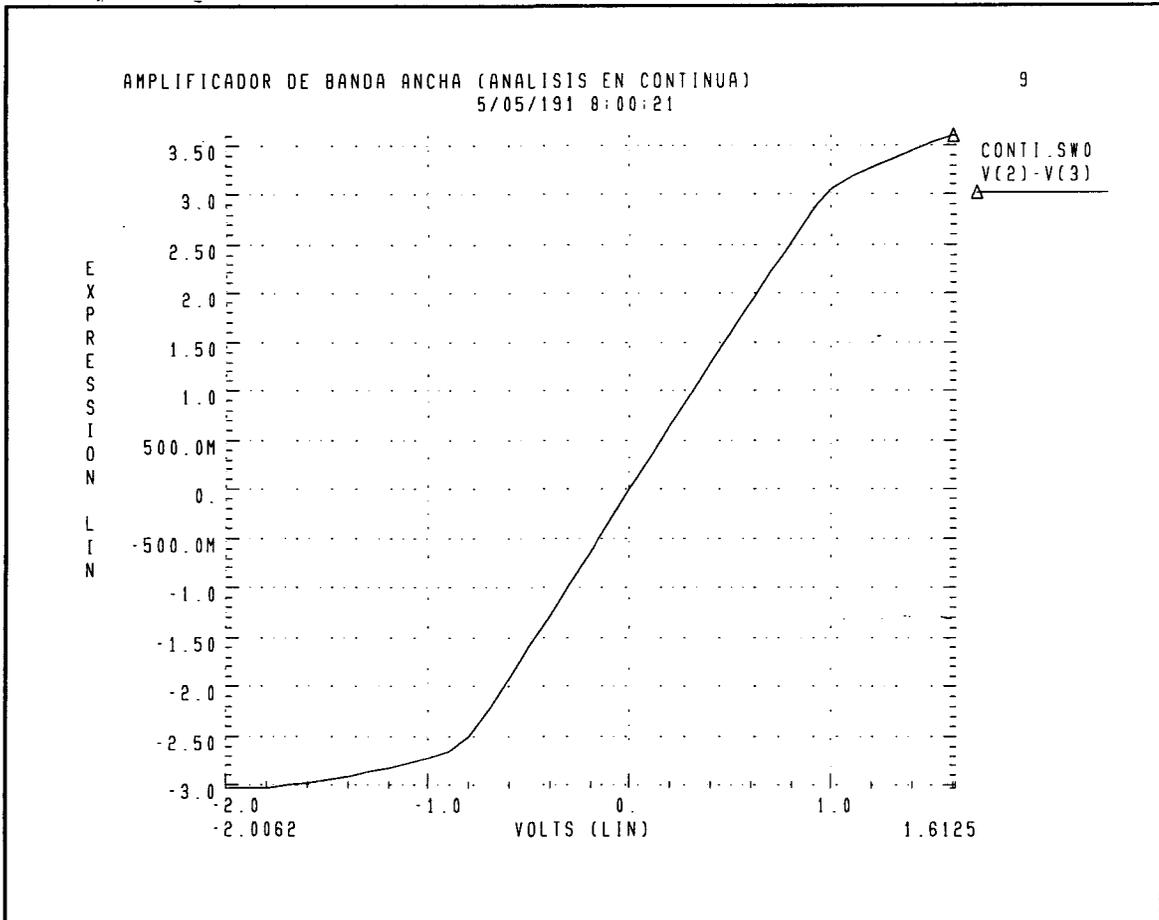
Análisis transitorio del Amplificador de Video.



Impedancia de entrada del Amplificador de Video.



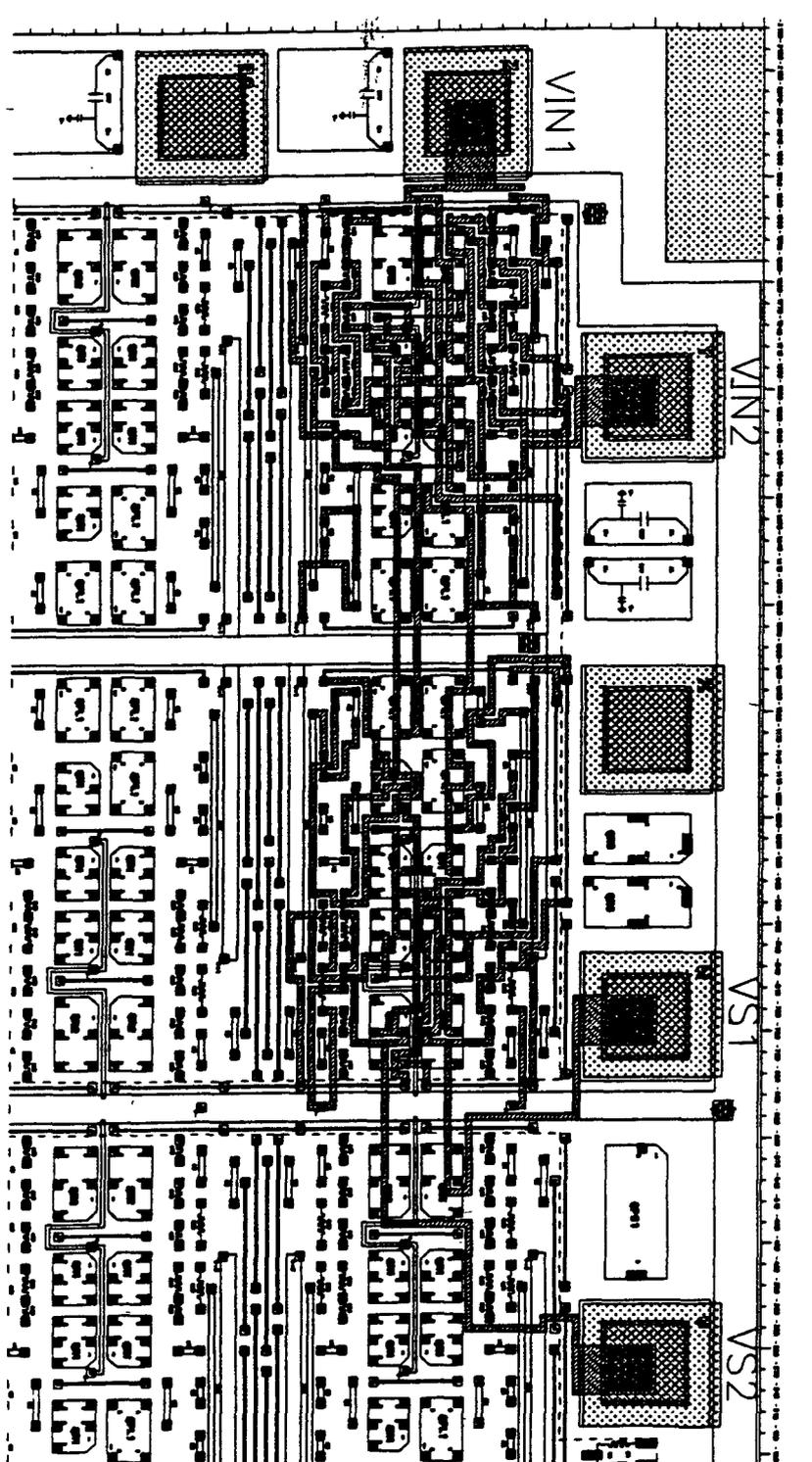
Respuesta en frecuencia del Amplificador de Video.



La potencia del amplificador de banda ancha diseñado es de 133,5 mW. El área utilizada para la realización de su layout es de dos tiles (aproximadamente 600\*450 micras).

Todas las simulaciones realizadas son para una configuración concreta, pudiendo variar ésta según los intereses deseados. En la realización del layout cabe la posibilidad de incluir dos nuevas conexiones, las cuales hicieran posible un cambio de configuración desde el exterior. Simplemente se cortocircuitarían alguna de las resistencias

de emisor del par diferencial de entrada, para así variar la ganancia del circuito.



## **Características del Amplificador de Banda Ancha**

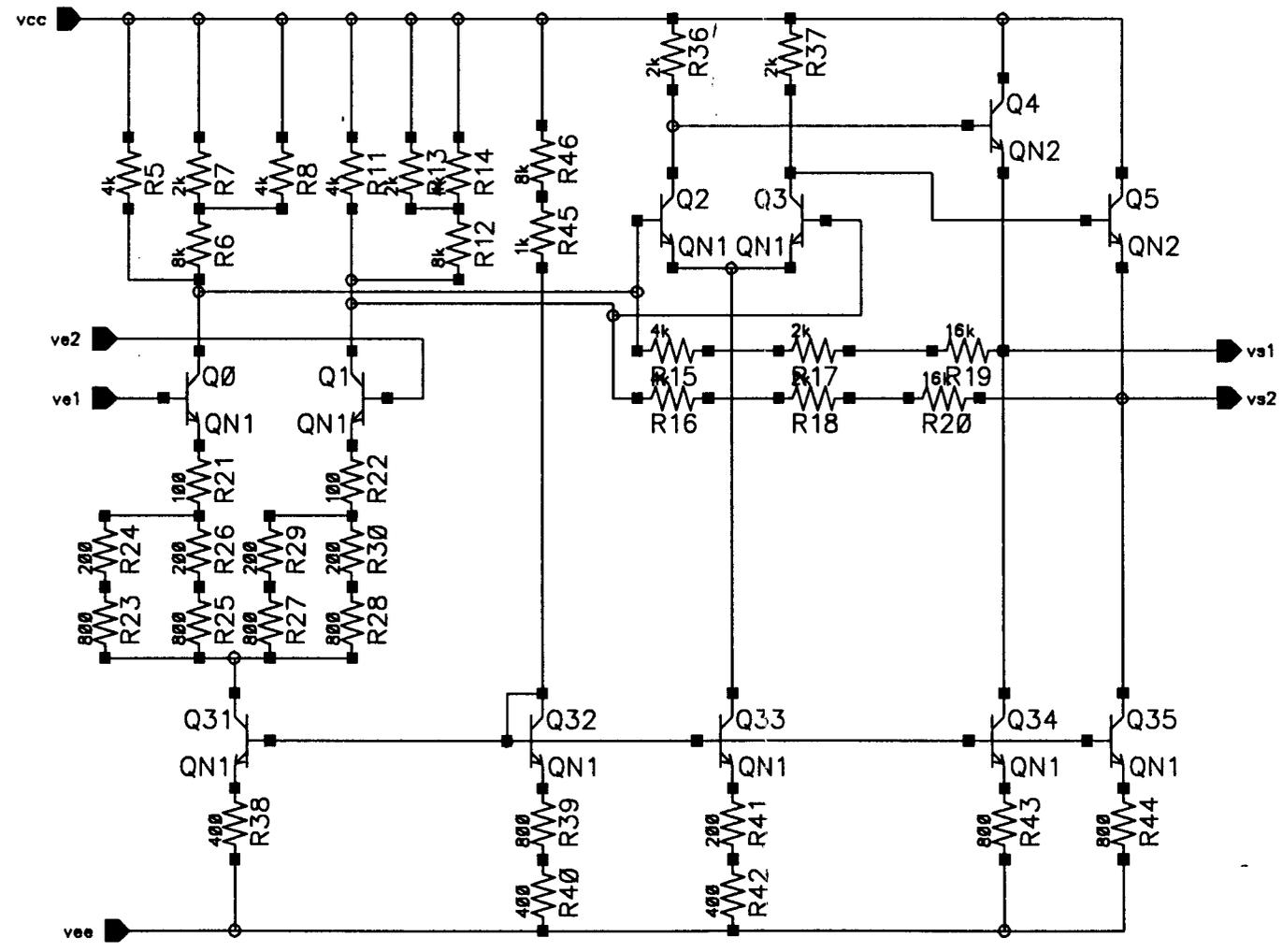
	<b>Amplificador de Banda Ancha</b>
<b>N° de Transistores</b>	<b>10</b>
<b>N° de Resistencias</b>	<b>35</b>
<b>Área Utilizada</b>	<b>2 tiles</b>
<b>Alimentación</b>	<b>+7,5 / -7,5 Voltios</b>
<b>Ganancia en dB</b>	<b>10,2 dB</b>
<b>Offset</b>	<b>0 Voltios</b>
<b>Ancho de Banda</b>	<b>60 MHz</b>
<b>Impedancia de Entrada (<math>Z_i</math>)</b>	<b>470 Kohmios</b>
<b>Impedancia de Salida (<math>Z_o</math>)</b>	<b>18 Ohmios</b>
<b>Potencia</b>	<b>133,5 mW</b>

Tecnología: Bipolar Polyuse L de Thomson  
Simulaciones: Hspice  
Fecha: Noviembre 1994

Observaciones: Diversas Configuraciones

FSCM NO.	DWG NO.	SH	REV
----------	---------	----	-----

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPROVED



APPROVALS	DATE	<h1>AMPLIFICADOR DE BANDA ANCHA</h1>			
DRAWN					
CHECKED					
CHECKED					
CHECKED					
ISSUED		SIZE A	FSCM NO.	DWG NO.	REV
		SCALE	SHEET		OF

### 5.3.- Multiplicador.

Esta célula deberá ser capaz de controlar el valor de la ganancia, dentro de unos márgenes, con una señal que le podremos aplicar exteriormente. De esta manera la célula va a tener bastante flexibilidad a la hora de incluirla en cualquier circuito.

Para su diseño partimos de la idea de tomar dos pares diferenciales en los cuales controlaremos sus fuentes para hacer que funcionen o nó, o que funcionen con mayor o menor ganancia. Introduciendo la señal diferencial en las bases de los dos pares diferenciales y utilizando cargas comunes podemos lograr que una variación en la base de las fuentes se logre reflejar en la salida de la célula. Por lo tanto el diseño se basará en realizar un modulador, de tal forma que introduciendo una señal (moduladora) , la señal de salida será segun sea la moduladora (portadora).

Una primera idea de lo que debe ser este circuito es la representada en la siguiente figura 5.3.1:

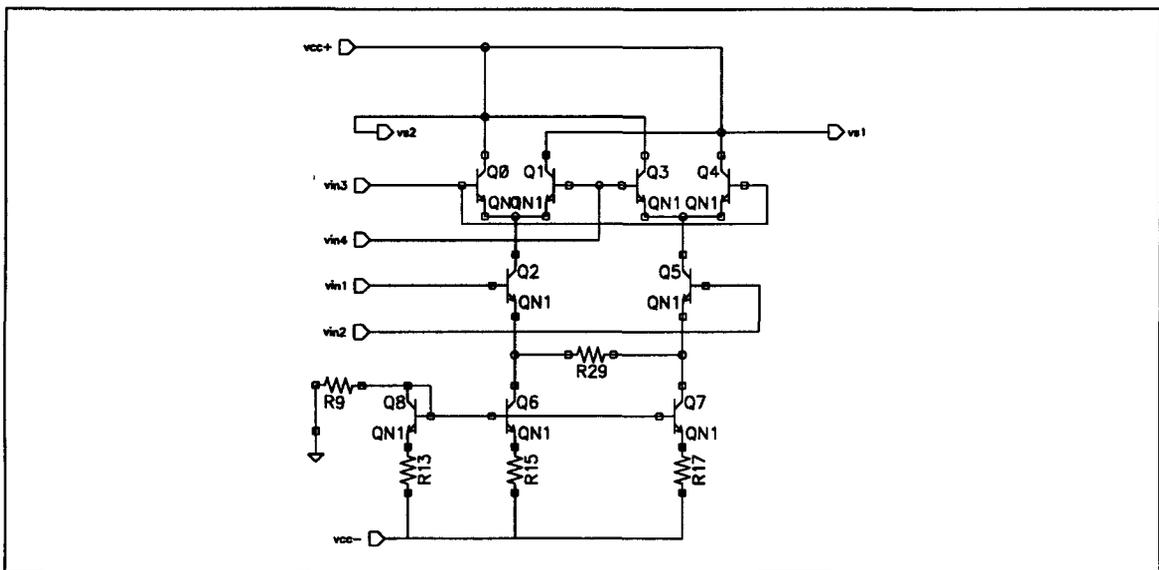


Figura 5.3.1.- Esquema inicial.

A continuación debemos adecuar el circuito para el funcionamiento que nosotros deseamos que realice. En primer lugar añadimos dos cargas iguales en los colectores de los pares diferenciales. Nuestro diseño será tal que la señal de entrada va a ser asimétrica por lo tanto una de las entradas del par diferencial habría que llevarla a cero.

En una de las fuentes debemos hacer que conduzca más o menos según deseemos mayor ó menor ganancia. Por lo tanto hacemos llegar una señal de rampa para comprobar el funcionamiento. La otra fuente la ponemos a cero.

Si intentamos hacer funcionar el circuito de esta forma podremos ver que la salida no es la deseada. Esto es debido a que el punto en continua del circuito no es correcto, y es que si nos fijamos la señal a la entrada no puede tener 0 voltios ya que de esta forma el transistor de la fuente estaría cortado. Por lo tanto debemos hacer que la señal de entrada tenga, según nuestros cálculos un nivel en continua de al menos 3 voltios. Debemos tener en cuenta que la segunda entrada por lo tanto deberá estar también con un nivel de 3 voltios en continua.

Con esta configuración obtenemos unos márgenes de funcionamiento tales que:

Tensión de pico a la entrada: 20 mV.

Salida:	control	nivel salida
	0 V.	0 V.
	0.5 V.	290 mV.
	1 V.	620 mV.

1.5 V. 800 mV.

Tension de pico a la entrada: 30 mV.

Salida: control nivel de salida maximo

0.8 V. 800 mV.

Observamos que el mayor nivel de salida va a estar en torno a los 800mV. y que si queremos mayor margen de precisión debemos hacer que la entrada sea de menos de 20mv ya que asi la señal de control tiene mayor margen de variación hasta alcanzar el nivel

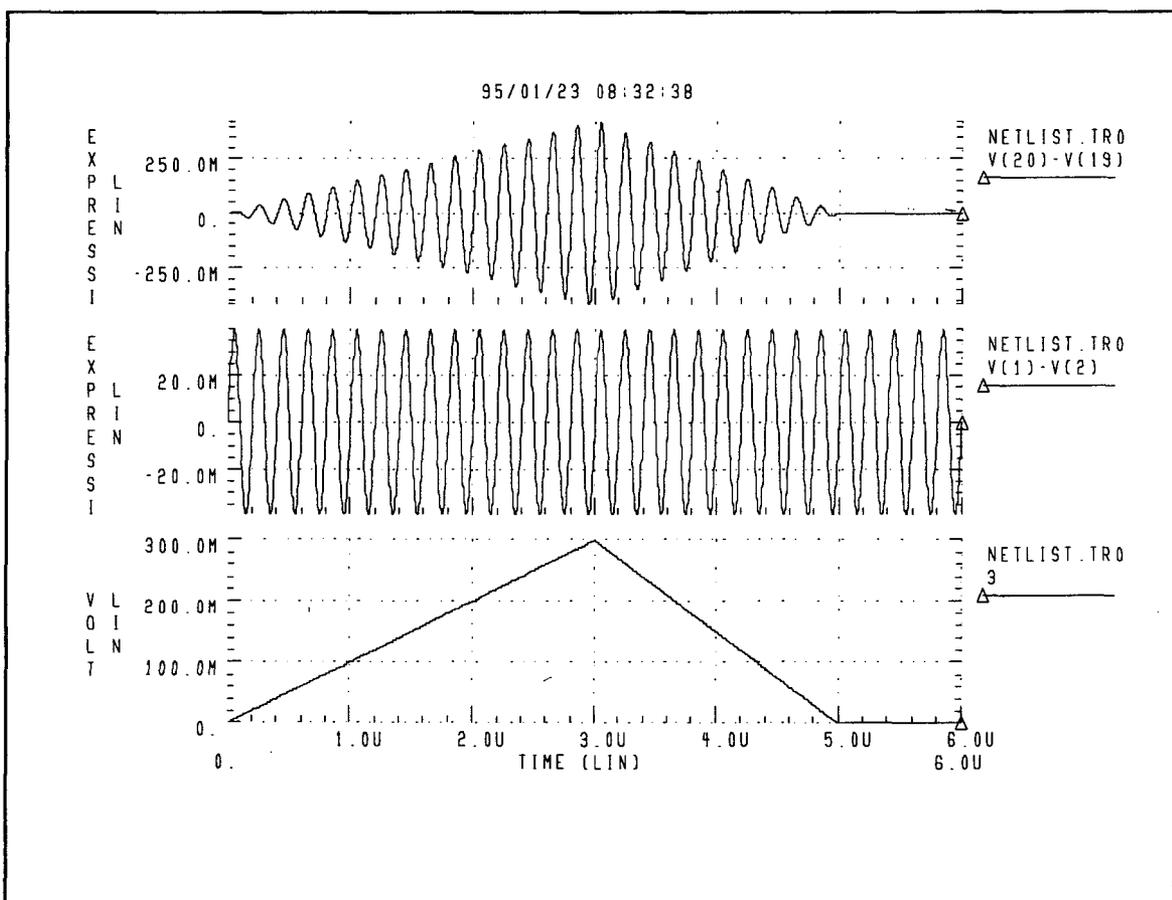


Figura 5.3.2.- Regimen transitorio del Multiplicador.

máximo a la salida.

A continuación se muestran las respuestas en régimen transitorio, el ancho de banda y la impedancia de entrada y salida. En la figura donde se muestra el transitorio podemos observar la señal de entrada, la de salida y la señal con la que se ha controlado. La impedancia de entrada es de 1k2 y la de salida de unos 900 Ohmios. Para calcular el ancho de banda hemos fijado la señal de control y hemos estudiado la célula como un simple amplificador.

La potencia disipada por esta célula es de 94,5 mW. El área usada para el diseño de su *layout* es de 2 tiles, 600\*450 micras.

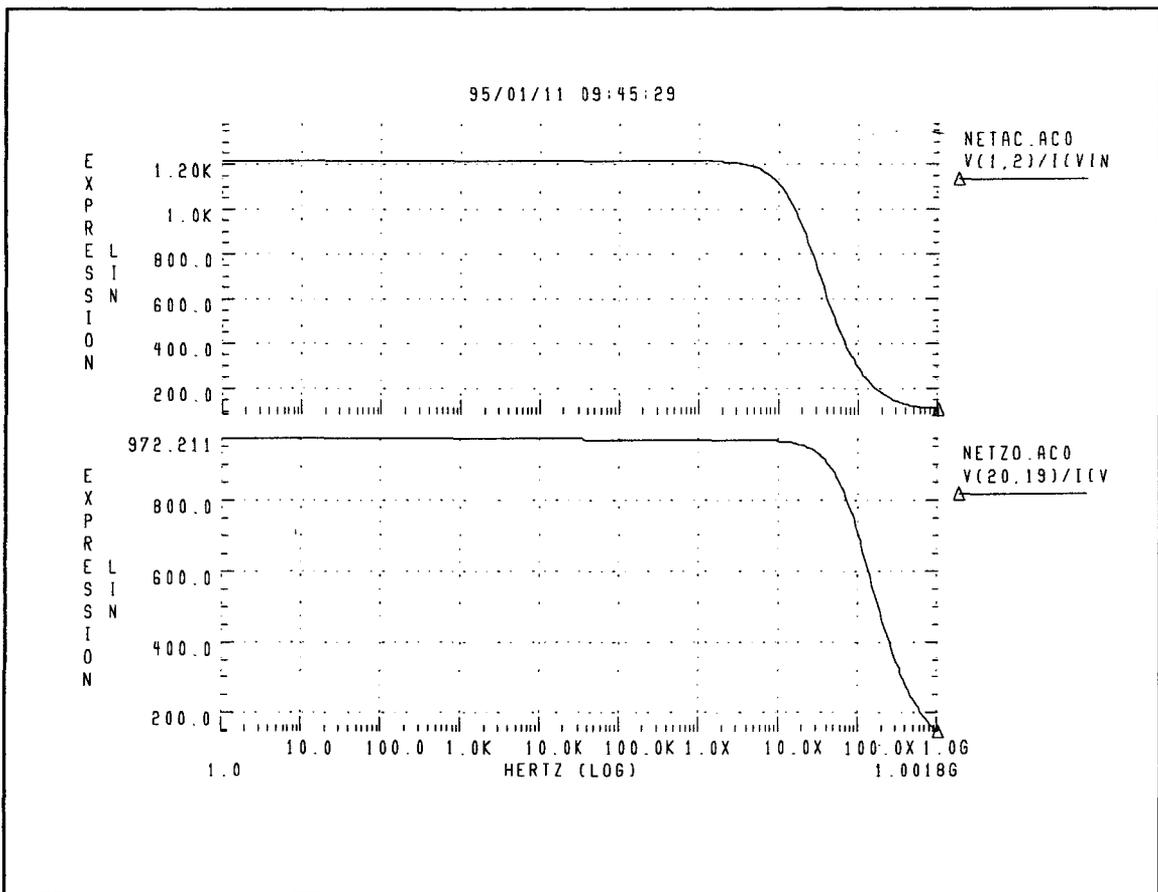


Figura 5.3.4.- Impedancia de entrada y salida del Multiplicador.

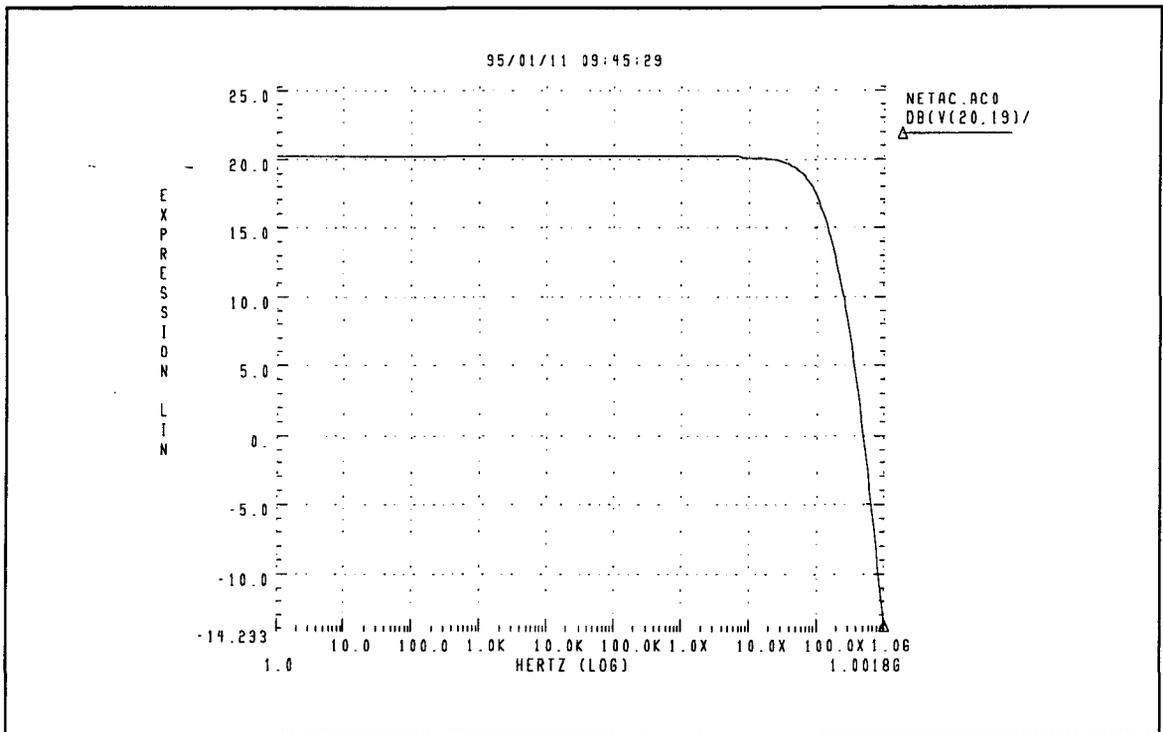


Figura 5.3.3.- Respuesta en frecuencia del Multiplicador.

## **Características del Multiplicador** **Control de Ganancia**

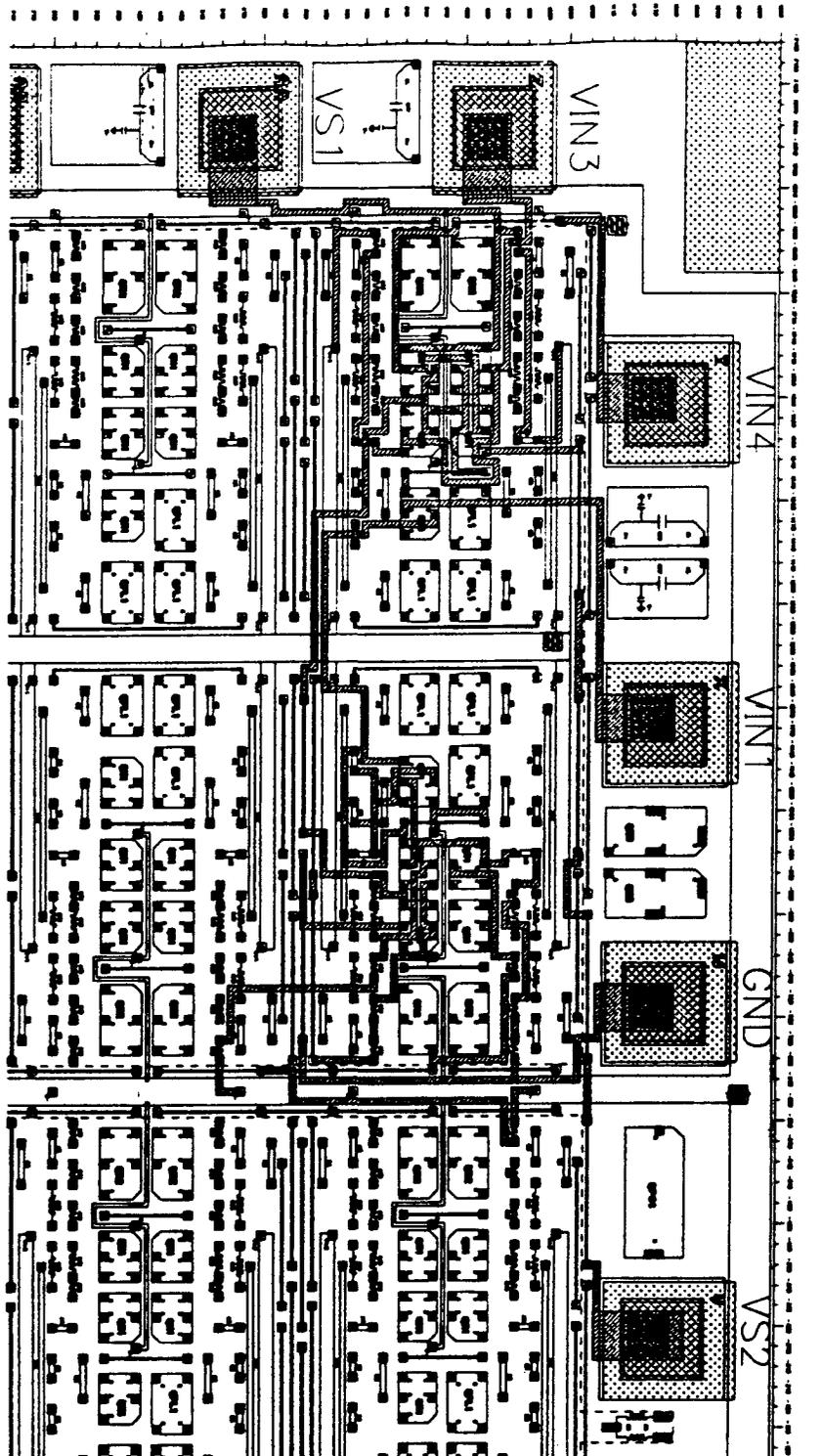
	<b>Multiplicador Control de AV</b>
<b>Nº de Transistores</b>	<b>9</b>
<b>Nº de Resistencias</b>	<b>13</b>
<b>Área Utilizada</b>	<b>2 Tiles</b>
<b>Alimentación</b>	<b>+7,5 / -7,5 Voltios</b>
<b>Ganancia en dB</b>	<b>Variable</b>
<b>Offset</b>	<b>0 Voltios</b>
<b>Ancho de Banda</b>	<b>107 MHz</b>
<b>Impedancia de Entrada (<math>Z_i</math>)</b>	<b>1,2 Kohmios</b>
<b>Impedancia de Salida (<math>Z_o</math>)</b>	<b>800 Ohmios</b>
<b>Potencia</b>	<b>95,7 mW</b>

Tecnología: Bipolar PolyuseL de Thomson

Simulaciones: Hspice

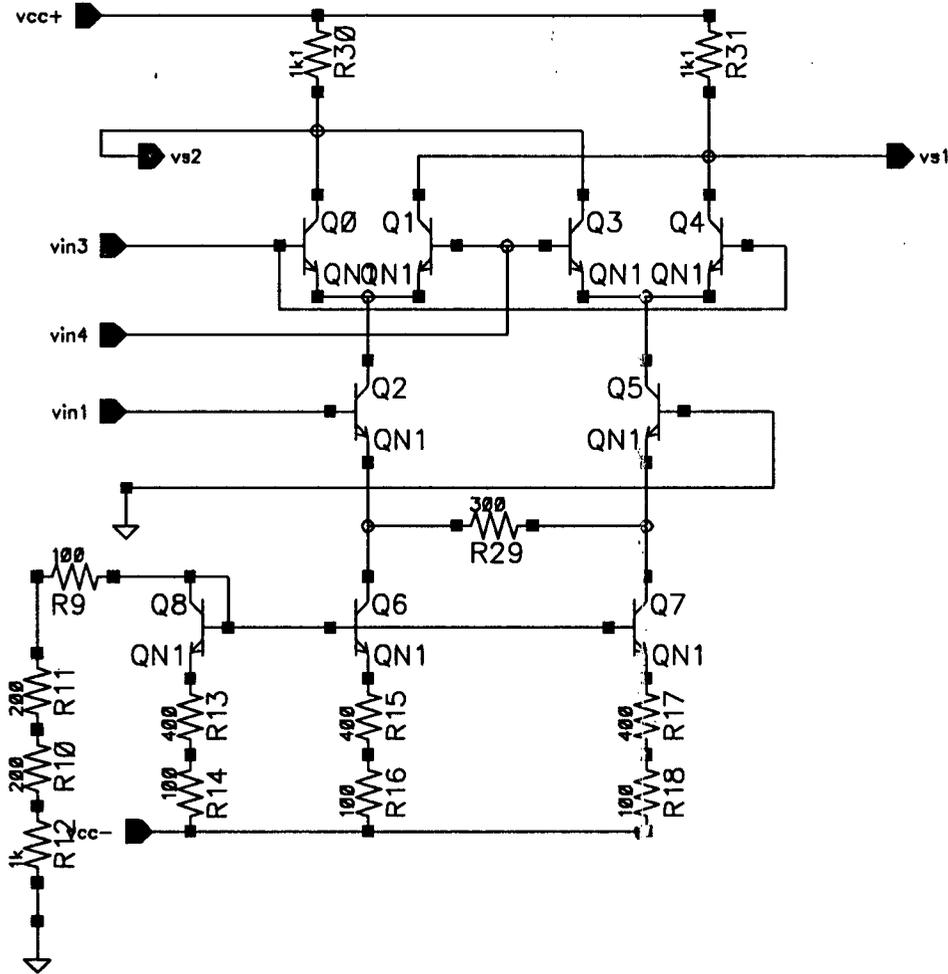
Fecha: Noviembre 1994

Observaciones: Configurado para funcionar como control de ganancia utilizando una señal exterior.



FSCM NO.	DWG NO.	SH	REV
----------	---------	----	-----

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPROVED



APPROVALS		DATE	CONTROL DE GANANCIA			
DRAWN						
CHECKED						
CHECKED						
CHECKED						
ISSUED			SIZE A	FSCM NO.	DWG NO.	REV
			SCALE			SHEET OF

### 5.4.- Multiplexor de dos entradas.

El diseño de esta célula ha surgido de la necesidad de seleccionar una de entre N entradas de señal y transmitir la señal seleccionada a un canal único. Para poder obtener una señal de varias disponibles diseñaremos una célula capaz de seleccionar la entrada deseada utilizando una señal de control. Un posible diagrama de bloques de esta célula consta de una etapa de control, otra de transmisión a canal único, y por ultimo, una serie de referencias para las fuentes de corrientes y etapa de control.

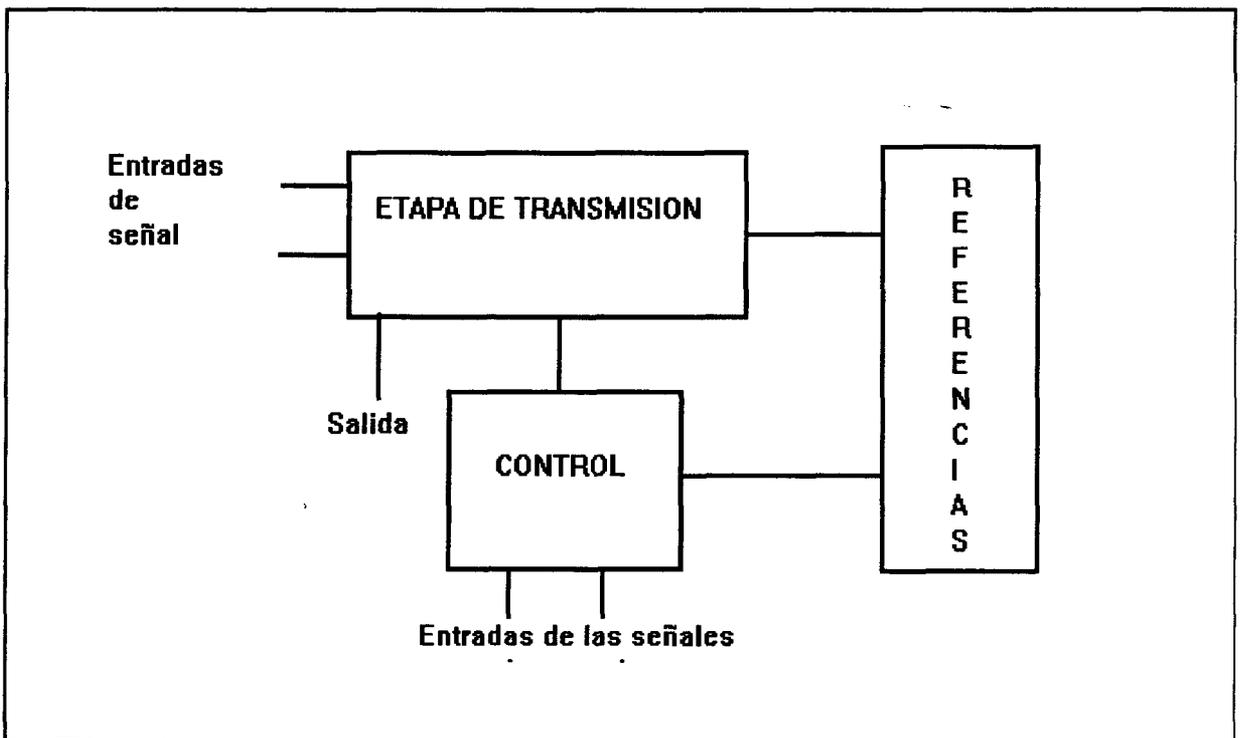


Figura 5.4.1.- Estructura de bloques usados en el multiplexor.

La denominada etapa de control consta de un par diferencial con transistores *pnp*, al cual le llega la señal de control por una de sus entradas. La otra entrada tiene una

tensión fija. El funcionamiento será de la siguiente manera: cuando a la entrada no tengamos tensión en el colector de q1 habrá mayor nivel de tensión que en el colector de q2. De esta manera si tomamos la salida en los colectores obtendremos una señal cuadrada según sea la señal de control que introduzcamos.

Esta señal cuadrada obtenida es utilizada por la etapa que denominamos de transmisión a canal único de la siguiente manera. Llevando la señal a la base de los transistores que funcionan como fuente de corriente lograremos que solo funcione una etapa según sea la señal. Si a la entrada del par diferencial de esta etapa hemos llevado las dos señales a multiplexar solo obtendremos a la salida una de ellas según sea nuestra señal de control.

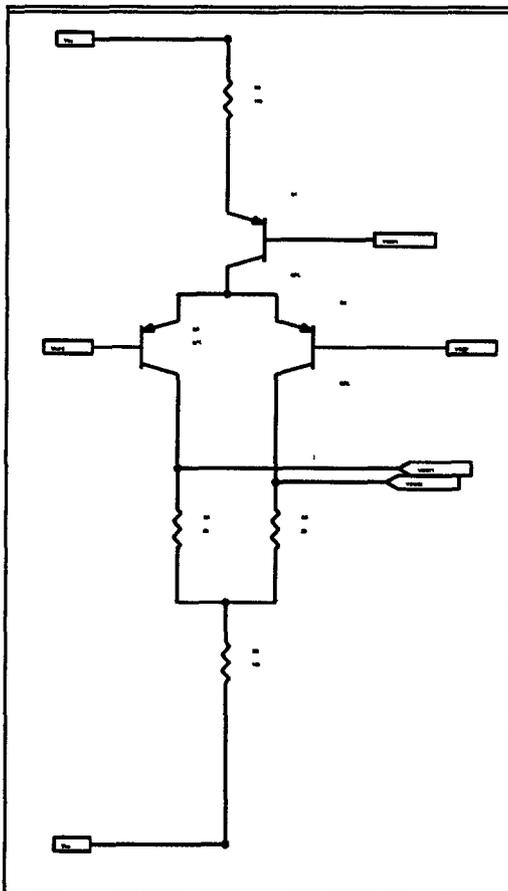


Figura 5.4.2.- Etapa de control.

Veamos las distintas etapas por separado para un estudio más completo de ellas.

a) Etapa de control. Como ya hemos comentado consta de un par diferencial *pnp* polarizado en continua por una referencia formada por q10, q30 y R. En el q19 hemos introducido la señal de control que va a oscilar entre 0 y 5 voltios. De esta manera los transistores quedarán cortados alternativamente logrando señal continua en los colectores alternativamente también. La referencia fija una intensidad en la fuente (q4) y un nivel de

1.444 voltios en la base del transistor q34. De esta manera cuando en la base de q19 tenemos 0 voltios éste fija una tensión en el emisor igual a la tensión base-emisor 0.722 voltios. Con esta tensión de emisor el transistor q34 queda cortado debido a que su diferencia de tensión entre base y emisor es muy pequeña. En este caso obtenemos una tensión de -3.8 voltios en el colector de q19 y una tensión de -3.5 en el colector de q34. Si a la entrada de q19 tenemos 5 voltios q34 fijara la tensión en el emisor y q19 quedará cortado. Los niveles de tensión en los colectores seran los mismos pero cambiados.

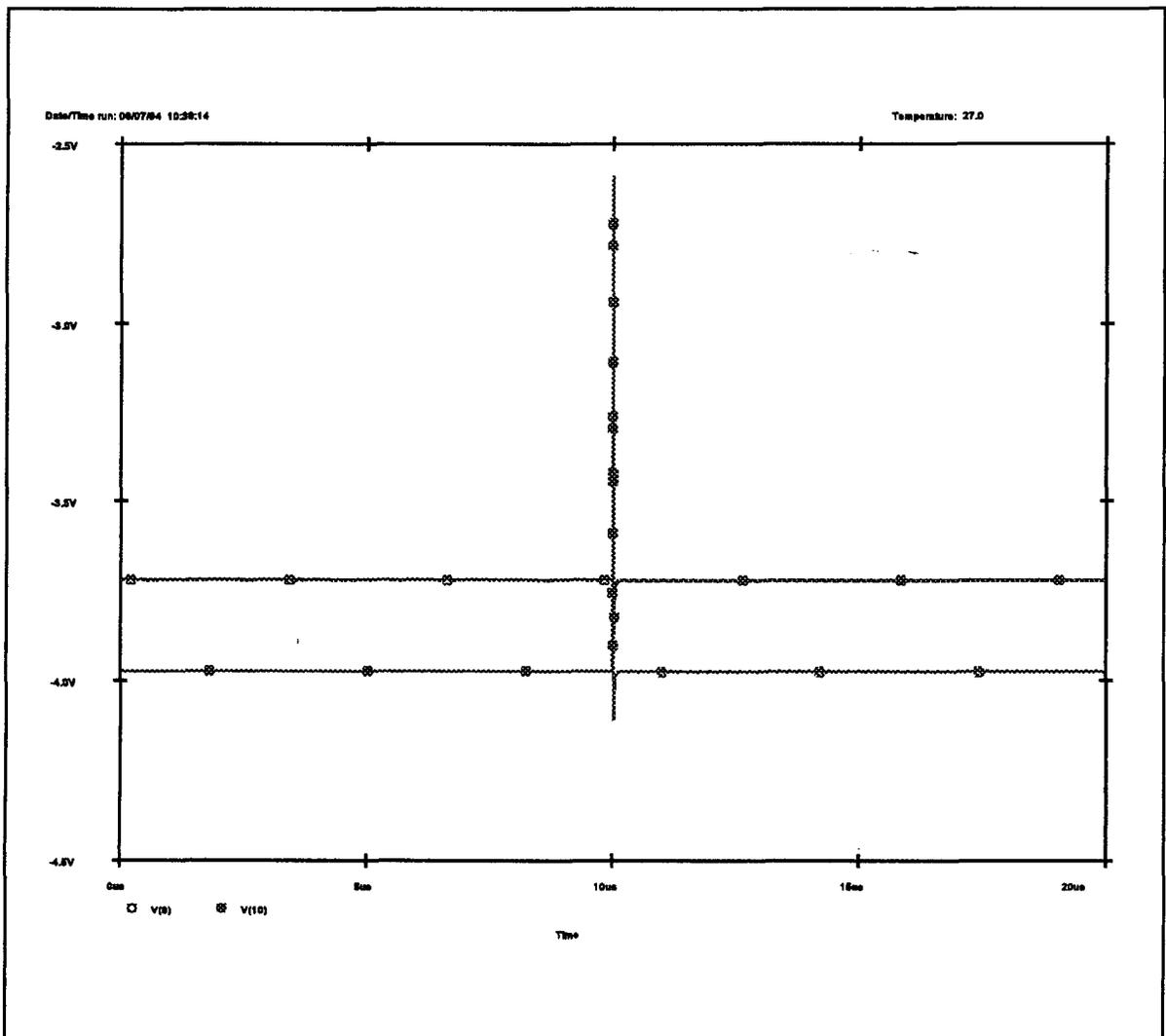


Figura 5.4.3.- Conmutación de los niveles de continua de la etapa de control.

b) Etapa de transmisión a canal único. Compuesta por dos pares diferenciales que comparten las cargas activas en sus colectores y sus respectivas fuentes. La señal que obtenemos de los colectores de la etapa de control es llevada una a cada una de las fuentes de tal forma que funcionaran alternativamente, según haya o no suficiente señal en su base. De ahí que los pares diferenciales compartan las cargas en sus colectores ya que nunca van a funcionar a la vez. De los colectores se toma la salida hacia un seguidor de emisor. Todo realimentado a las bases mediante un transistor.

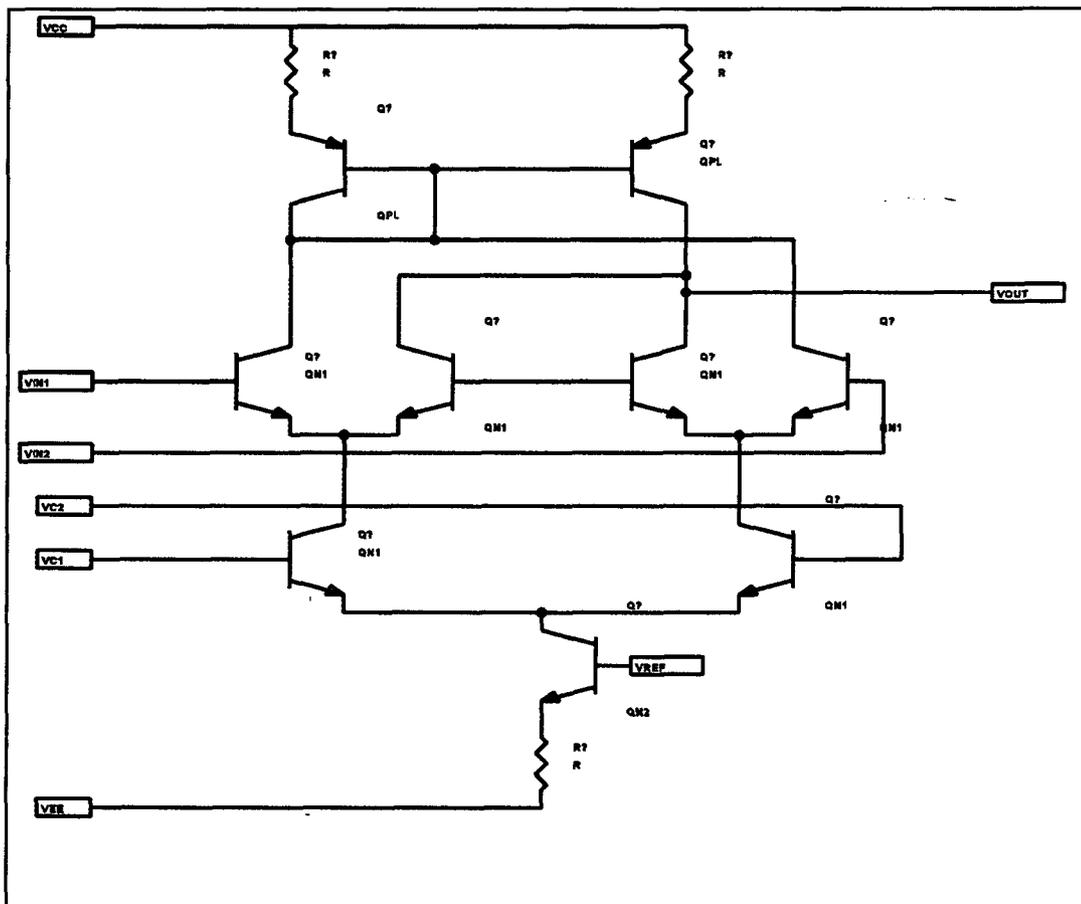


Figura 5.4.4.- Etapa de transmisión sin buffer.

Como a las bases de las fuentes nos va a llegar ó -3.5 ó -3.8 voltios siempre habrá uno que fije la tensión en el emisor quedando el transistor de la otra fuente cortado.

c) Referencias. Son espejos de corriente utilizados para obtener las referencias de las fuentes y otros puntos donde queramos fijar tensión. Segun sean estos valores funcionará el circuito. Con ellos no sería difícil realizar una restructuración del circuito para un menor consumo de potencia, un cambio de fuentes de alimentación, etc...

Por último vamos a llamar la atención sobre la similitud entre la etapa de control y la etapa de transmisión. Si nos fijamos estan compuestas por las mismas partes con la diferencia de que una es con transistores *pnp* y otra es con transistores *npn* y cargas activas en los colectores en vez de resistencias.

El multiplexor ha sido la célula mas complicada a la hora de realizar su layout, debido a las multiples referencias y pares diferenciales con cargas activas. El área utilizada ha sido de dos tiles, 600\*450 micras.

La potencia disipada por el multipexor es de 52,5 mW.

A continuación presentamos las distintas gráficas de las simulaciones realizadas sobre este diseño.

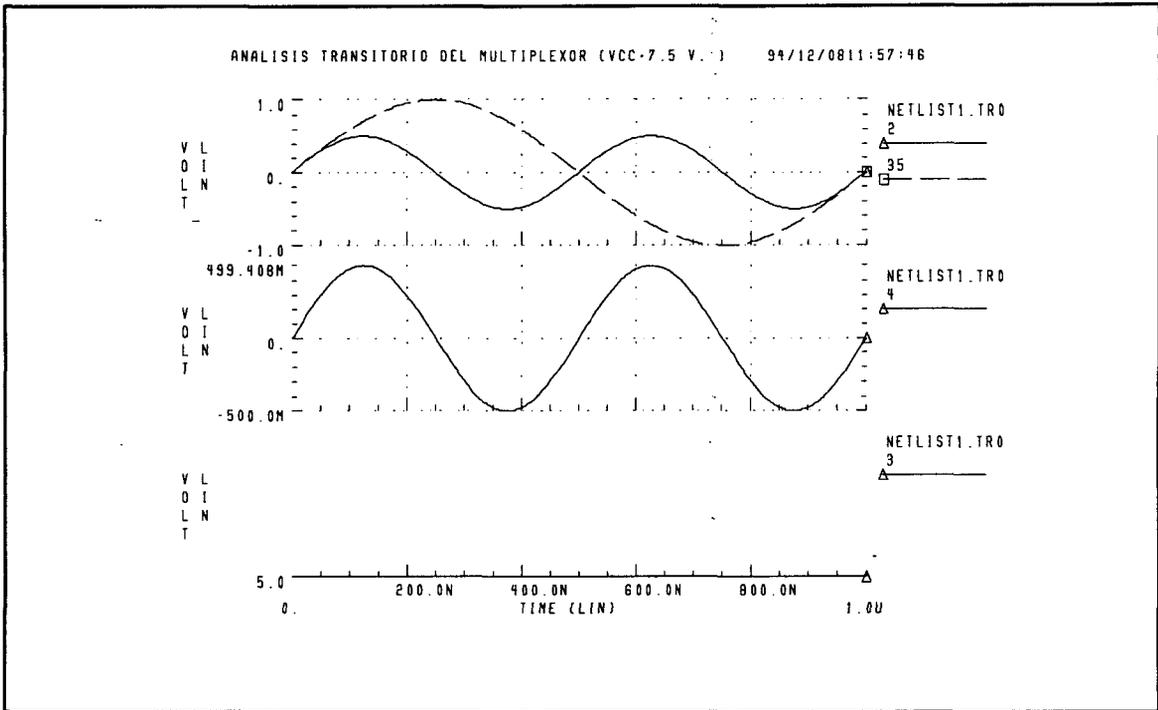


Figura 5.4.5.- Selección de una de las señales de entrada.

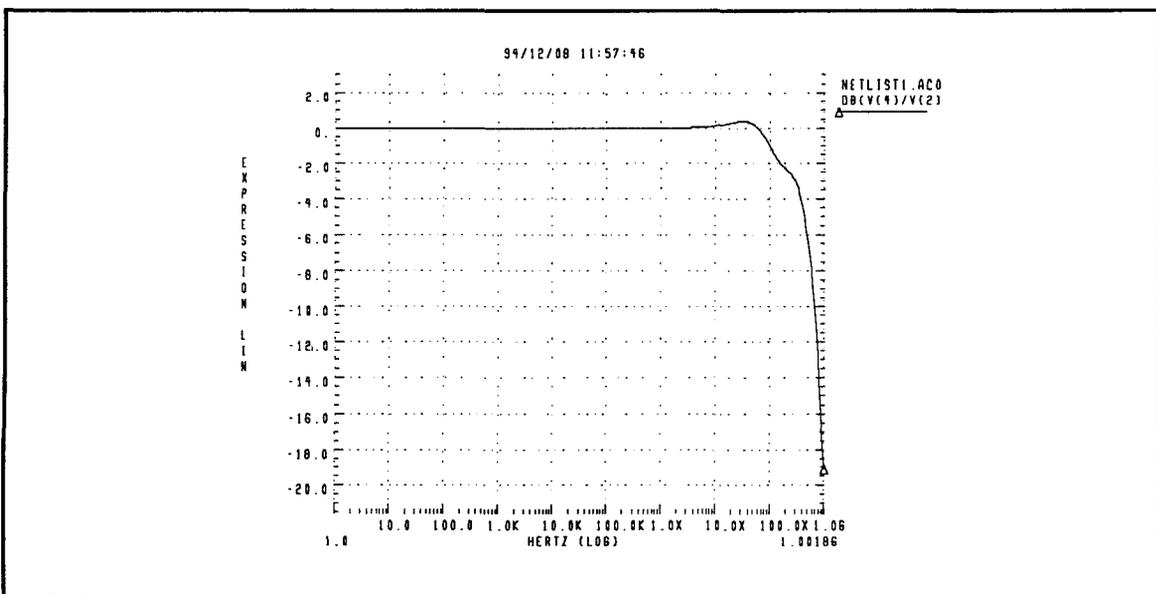


Figura 5.4.7.- Respuesta en frecuencia.

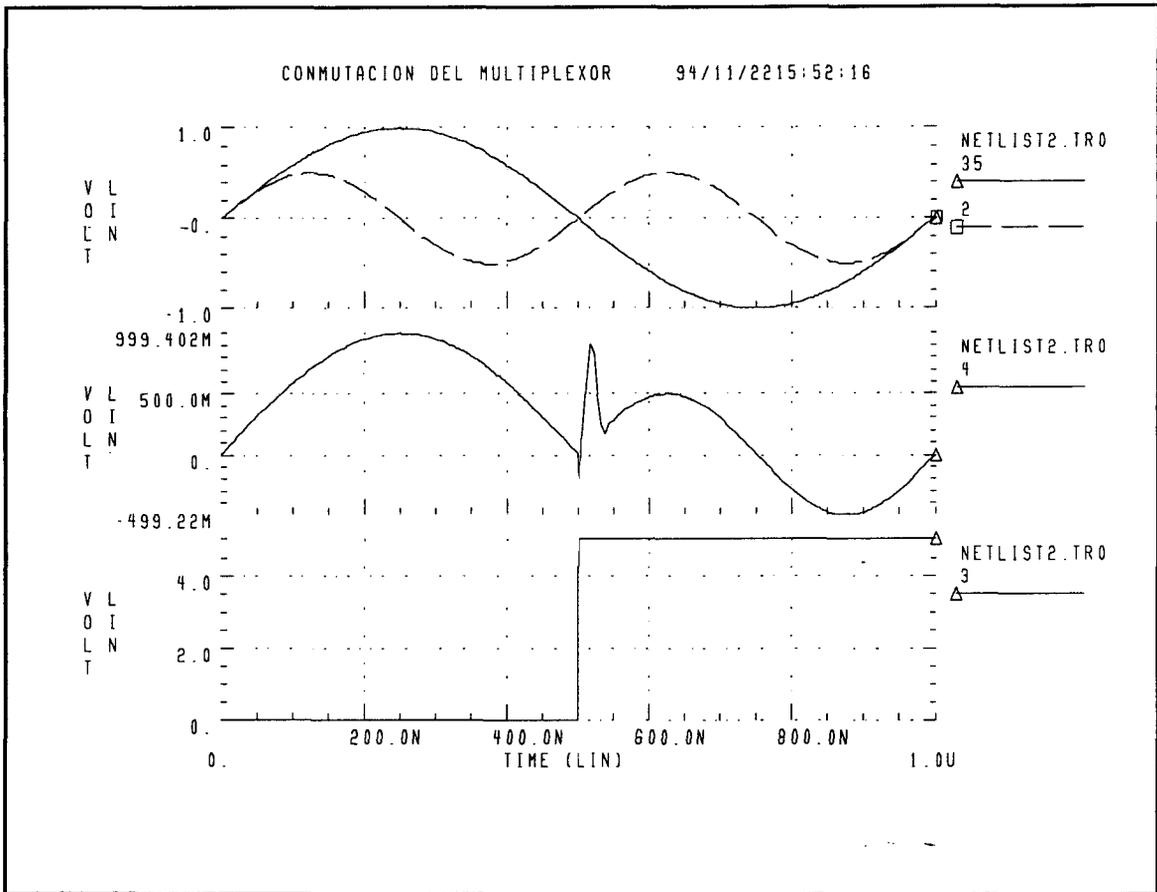


Figura 5.4.6.- Selección de las dos señales de entrada.

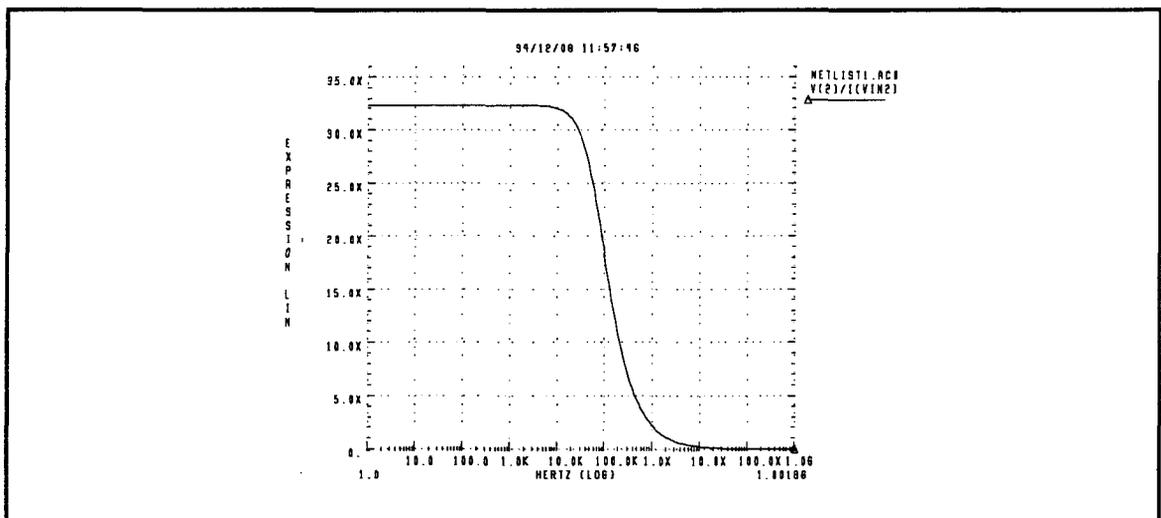
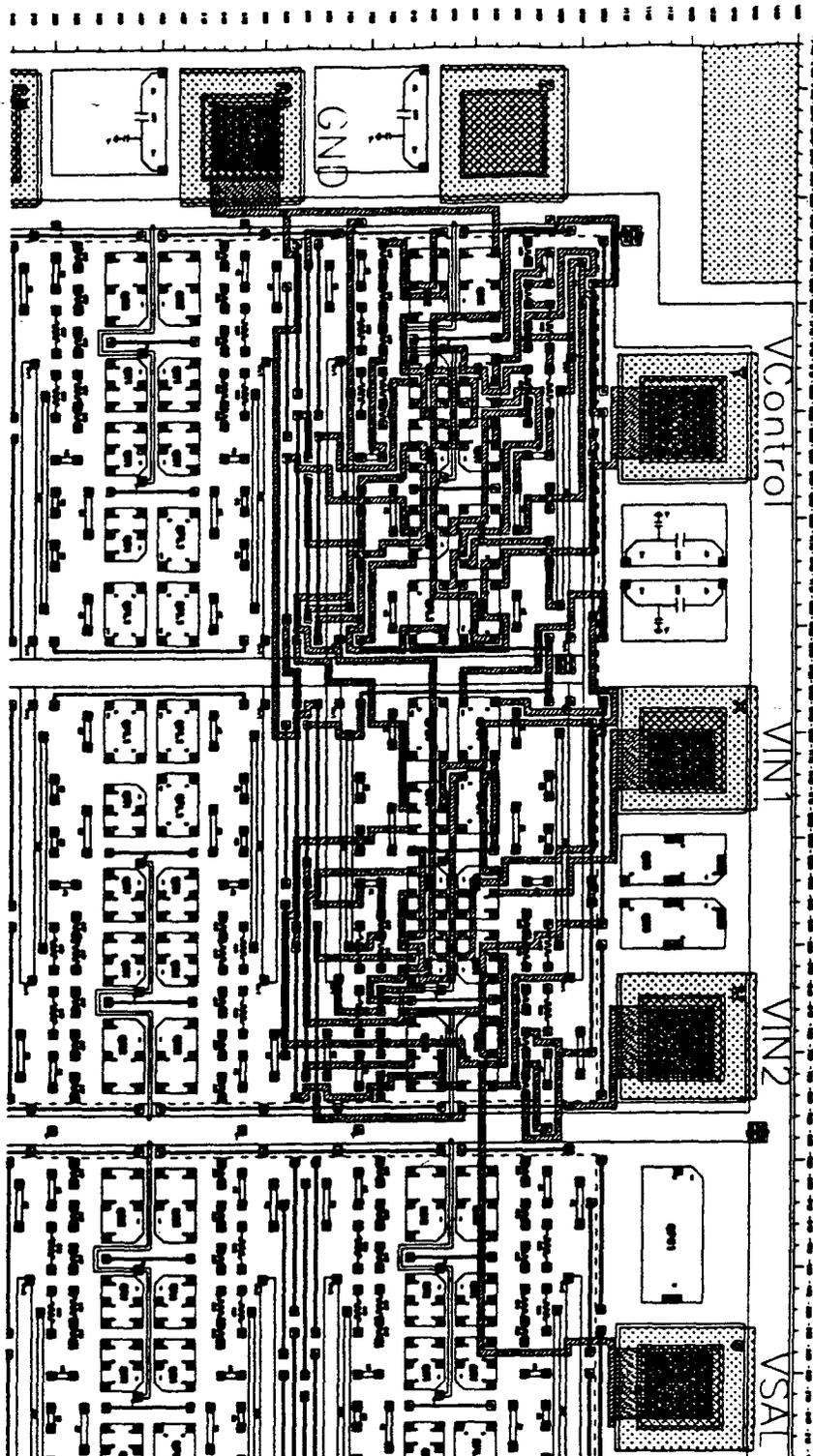


Figura 5.4.8.- Impedancia de entrada.



## Características del Multiplexor

	<b>Multiplexor</b>
<b>N° de Transistores</b>	<b>18</b>
<b>N° de Resistencias</b>	<b>22</b>
<b>Área Utilizada</b>	<b>2 Tiles</b>
<b>Alimentación</b>	<b>+7,5 / -7,5 Voltios</b>
<b>Ganancia en dB</b>	<b>1 dB</b>
<b>Offset</b>	<b>-1 mV</b>
<b>Ancho de Banda</b>	<b>300 MHz</b>
<b>Impedancia de Entrada (<math>Z_i</math>)</b>	<b>32 Kohmios</b>
<b>Impedancia de Salida (<math>Z_o</math>)</b>	<b>Nula</b>
<b>Potencia</b>	<b>4,25 mW</b>

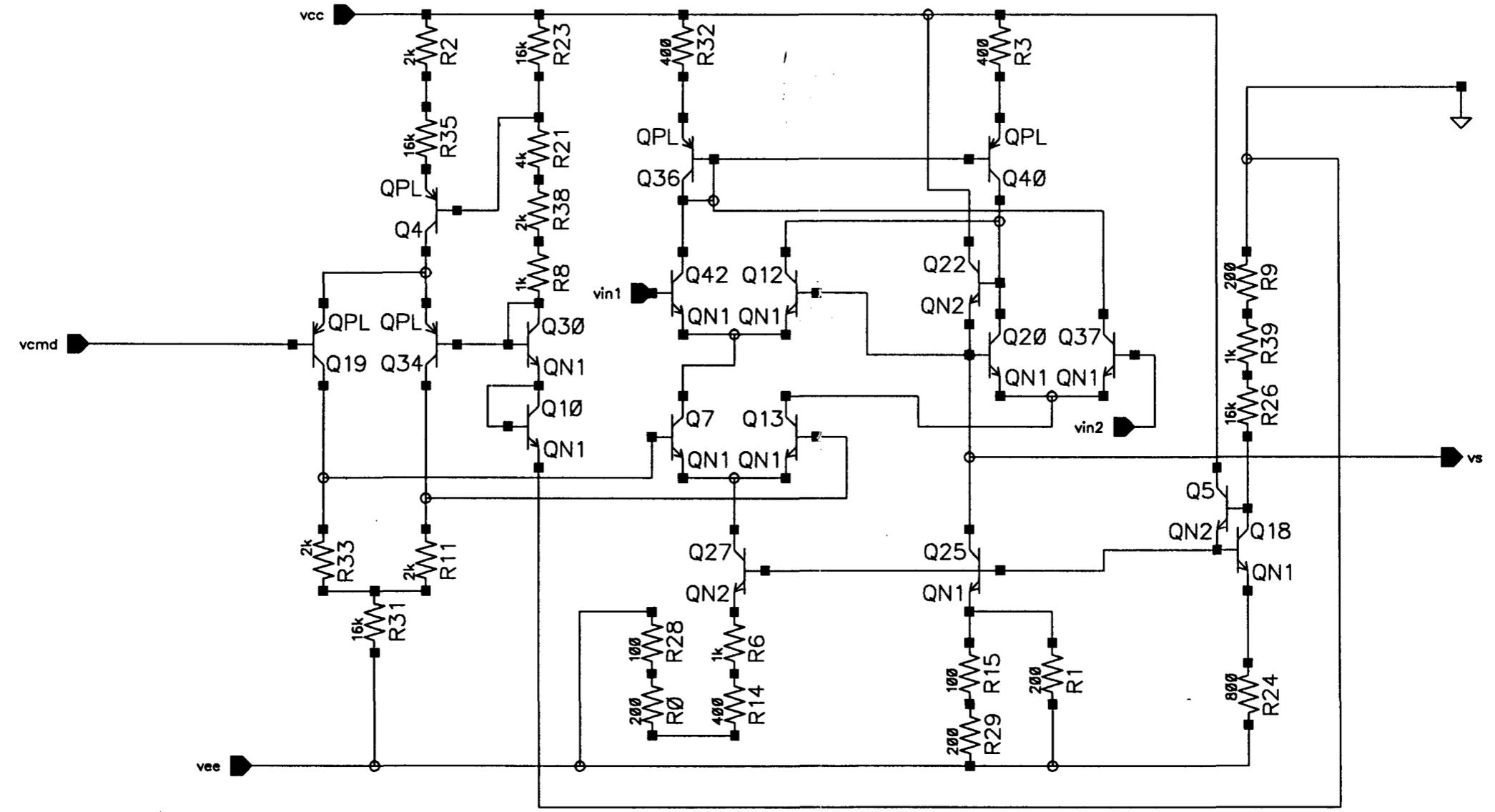
Tecnología: Bipolar PolyuseL de Thomson

Simulaciones: Hspice

Fecha: Octubre 1994

Observaciones: Diversas Configuraciones segun el Offset deseado.

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPROVED



APPROVALS	DATE
DRAWN	
CHECKED	
CHECKED	
CHECKED	
ISSUED	

MULTIPLEXOR  
2\_a\_1.(Tip.A)

SIZE <b>A</b>	FSCM NO.	DWG NO.	REV
SCALE	SHEET		OF



## **5.5. Operacional.**

El amplificador operacional posee las entradas por transistores *pnp*. El par diferencial formado por los transistores *pnp*, tienen en los colectores cargas activas. De esta manera al haber pequeñas variaciones de corriente se producen grandes variaciones de tensión debido a la alta impedancia de las cargas activas. El buffer de salida es un *push-pull*, el cual puede soportar de -6mA a 6mA. Se ha situado una rama donde se han situado dos diodos (Q19, Q20) para fijar una tensión de 1,4 Voltios y así evitar la distorsión de cruce en el amplificador clase AB de la salida. Los transistores Q36 y Q37 se utilizan formando un *pseudopnp* en vez de utilizar un *pnp*, para así, poder soportar mayor corriente a la salida. Del colector de Q1 se toma la salida hacia un seguidor de emisor y la salida de éste hacia un emisor común situado en la misma rama de los diodos. Un condensador de 6 pF suministra la compensación de estabilidad. El transistor tipo Qpl Q4 tiene cortocircuitada la base y el colector y funciona como referencia.

La célula puede ser usada con menor voltaje de alimentación (mínimo 6 Voltios).

La resistencia de la fuente de corriente debe ser corregida para mantener la corriente de bias si se realiza alguna variación. El consumo se puede reducir disminuyendo la corriente de bias. Variando esto, las características variarán.

En algunos casos, cualquier modificación del esquema deberá llevar consigo la caracterización de la nueva célula.

A la hora de realizar el *layout* de esta célula, hay que tener en cuenta que las entradas a los *pnp* deben ser idénticas, con transistores iguales y de la misma orientación.

El área que ocupa es de dos tiles, mas una capacidad en la periferia.

La potencia disipada por el operacional es de 42mW.

El offset es de 700 micro voltios. La relación de rechazo al modo común de 170 dB. La ganancia en lazo abierto es de 90 dB.

A continuación se muestran las distintas simulaciones realizadas sobre la célula, así como el esquema de éste.

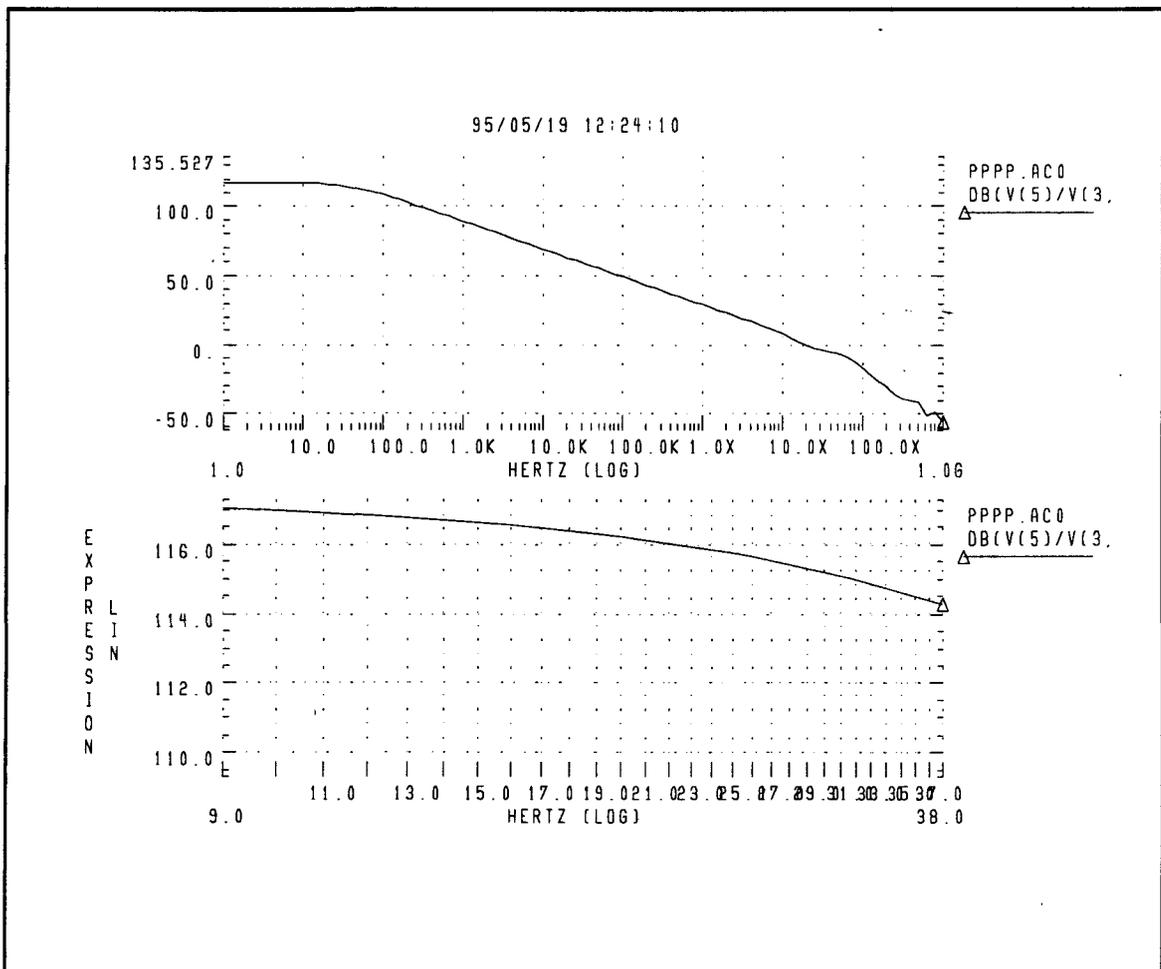


Figura 5.5.1.- Ganancia en Modo Diferencial.

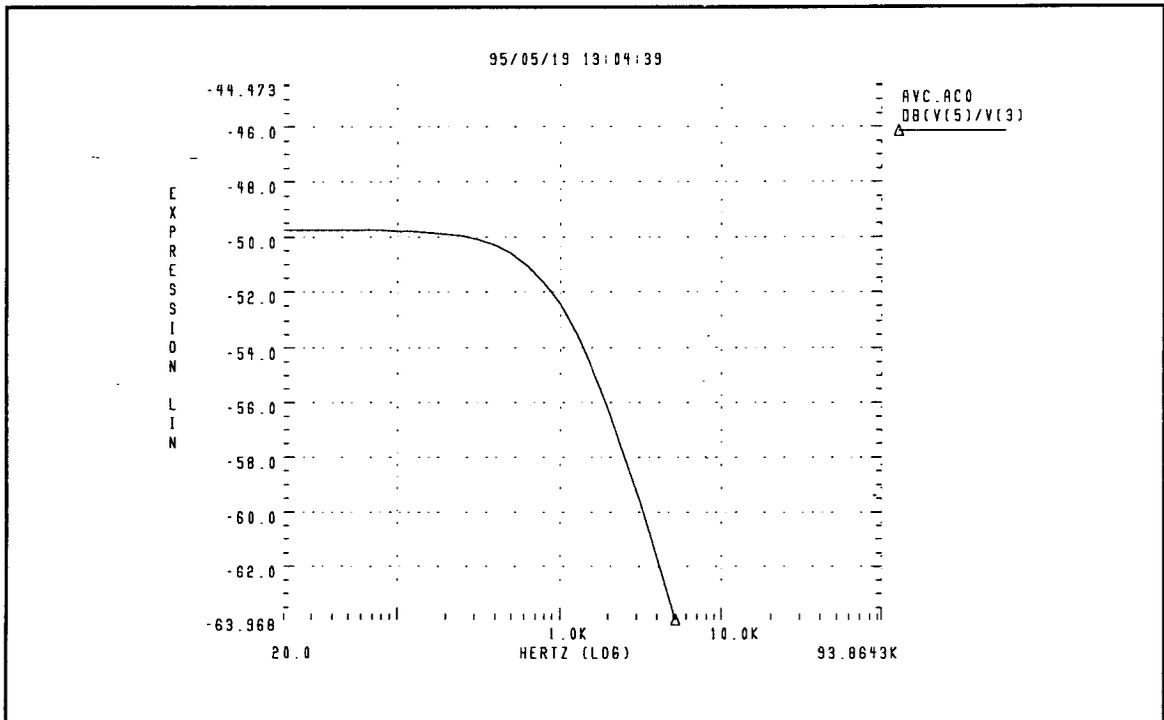


Figura 5.5.2.- Ganancia en Modo Común.

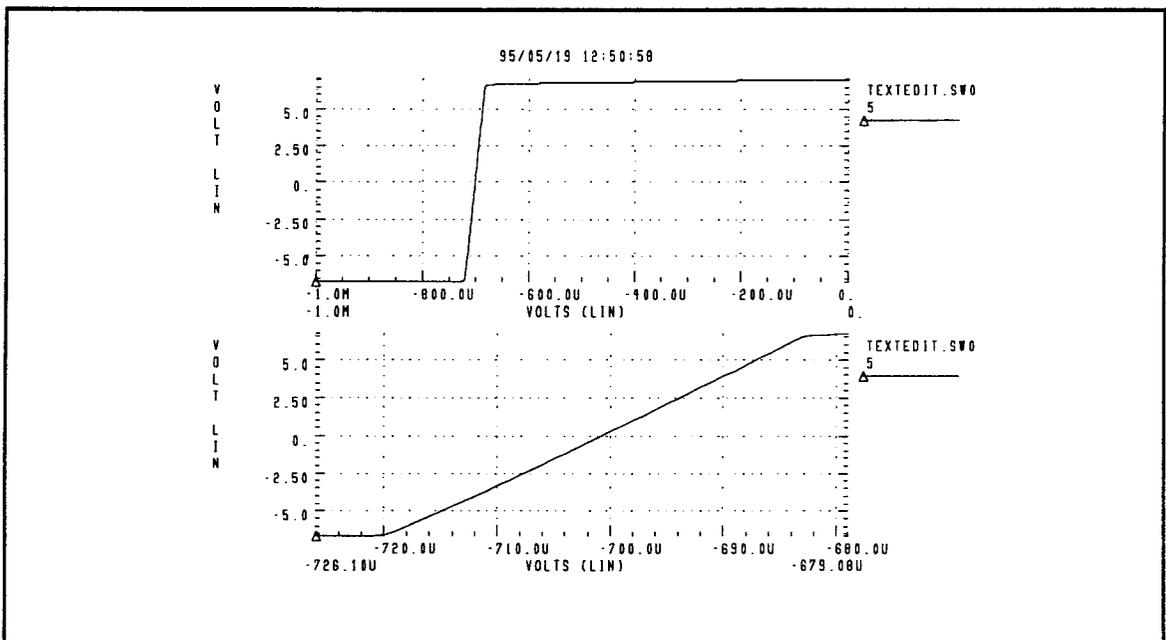


Figura 5.5.3.- Análisis en Continua. Medida del Offset.

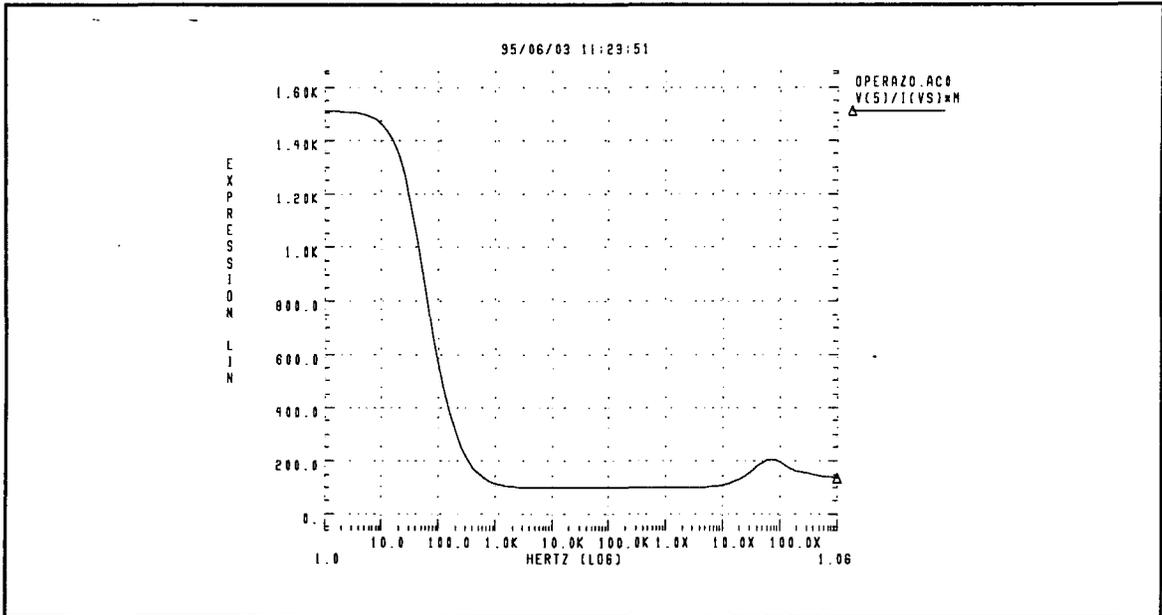


Figura 5.5.4.- Impedancia de Salida.

## Características del Amplificador Operacional

	<b>Amplificador Operacional</b>
<b>Nº de Transistores</b>	<b>15</b>
<b>Nº de Resistencias</b>	<b>24</b>
<b>Área Utilizada</b>	<b>2 Tiles</b>
<b>Alimentación</b>	<b>+7,5 / -7,5 Voltios</b>
<b>Offset</b>	<b>0,7 mV</b>
<b>Ganancia en Modo Común</b>	<b>-50 dB</b>
<b>Ganancia en Modo Dif.</b>	<b>116 dB</b>
<b>RRMC</b>	<b>166 dB</b>
<b>Ancho de Banda</b>	<b>40 Hz</b>
<b>Impedancia de Entrada (<math>Z_i</math>)</b>	<b>1,5 Kohmios</b>
<b>Impedancia de Salida (<math>Z_o</math>)</b>	<b>25 Kohmios</b>
<b>Potencia</b>	<b>4,2 mW</b>

Tecnología: Bipolar Polyuse L de Thomson

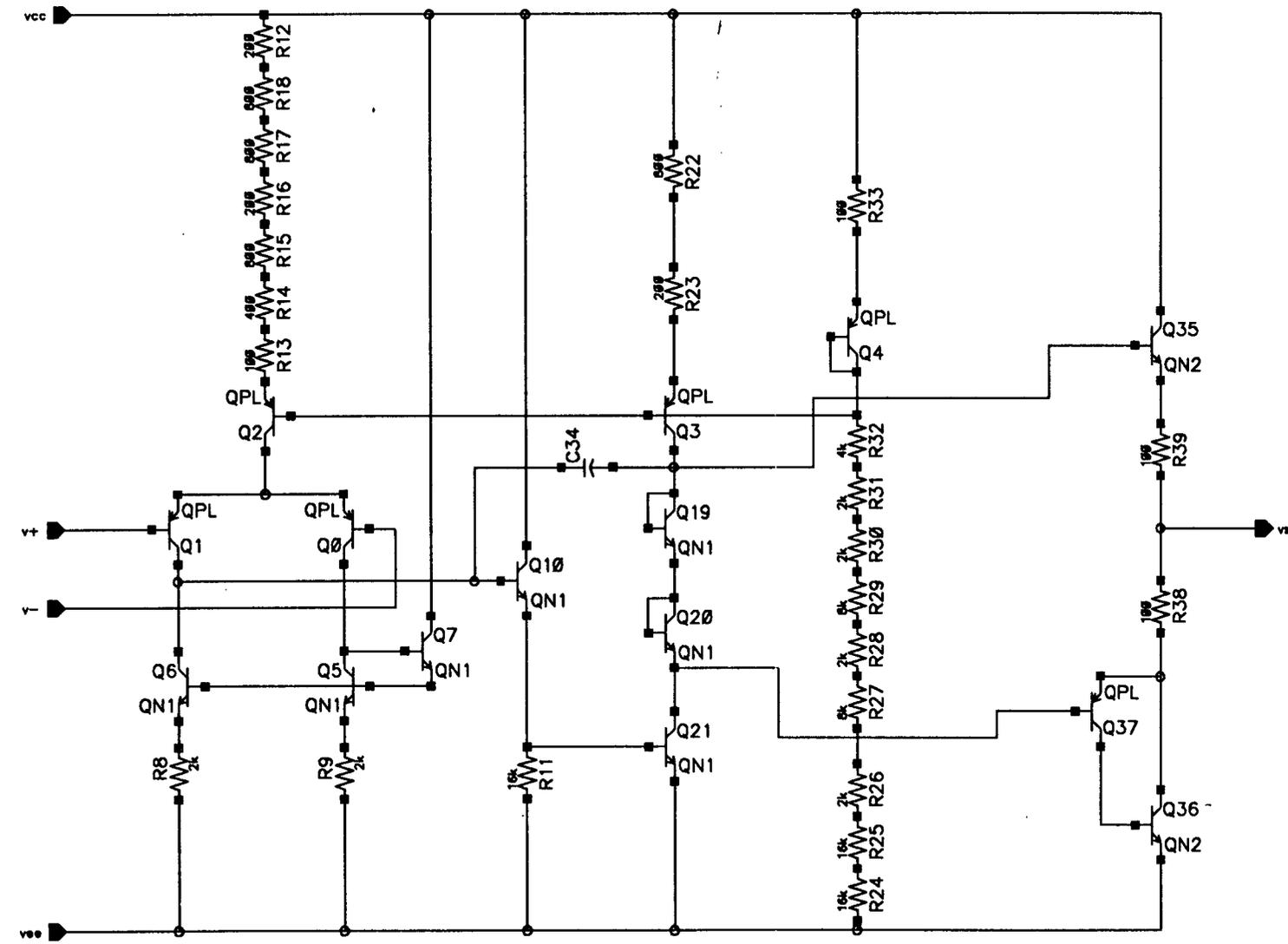
Simulaciones: Hspice

Fecha: Noviembre 1994

Observaciones: Uso de un Condensador de compensación.

FSCM NO.	DWG NO.	SH	REV
----------	---------	----	-----

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPROVED



APPROVALS	DATE	<h1>AMPLIFICADOR OPERACIONAL</h1>			
DRAWN					
CHECKED					
CHECKED					
CHECKED					
ISSUED		SIZE <b>B</b>	FSCM NO.	DWG NO.	REV
		SCALE	SHEET		OF

# Capítulo VI

## **Matriz de Conmutación con Control de Ganancia**

La aplicación que se ha realizado como muestra del diseño con la tecnología bipolar Polyuse L de Thomson es una matriz de conmutación. Con ella podremos seleccionar una entrada de varias señales disponibles y tras amplificarla la podremos obtener en la salida. Con el encapsulado disponible se puede hacer hasta de cuatro entradas. Para la realización de una de ocho entradas necesitaríamos un encapsulado mayor o dos de los que disponemos.

Para el diseño de esta matriz se han utilizado células diseñadas aparte: el amplificador de video, el multiplexor, el buffer o el multiplicador con las correspondientes modificaciones según su función. Además de estas células se han añadido una serie de estructuras con transistores y resistencias necesarias para el acondicionamiento de las otras.

- Multiplexor 4 a 1 analógico.

La primera parte de este diseño es la matriz de multiplexores. Esta parte es la encargada de seleccionar una señal. Para ello se han colocado tres multiplexores 2 a 1 de la forma que se muestra en la figura 6.1. Las salidas de los dos primeros se han llevado al tercero, del que se obtendrá la salida definitiva.

Como en posteriores etapas nos va a interesar que la señal tenga un cierto nivel de continua hemos rediseñado el valor de algunas resistencias de los multiplexores para ello. Concretamente se han cambiado el valor de una de las resistencias la referencia de 100 a 200 Ohmios, logrando con ello el desplazamiento de nivel continua deseado. Así podemos

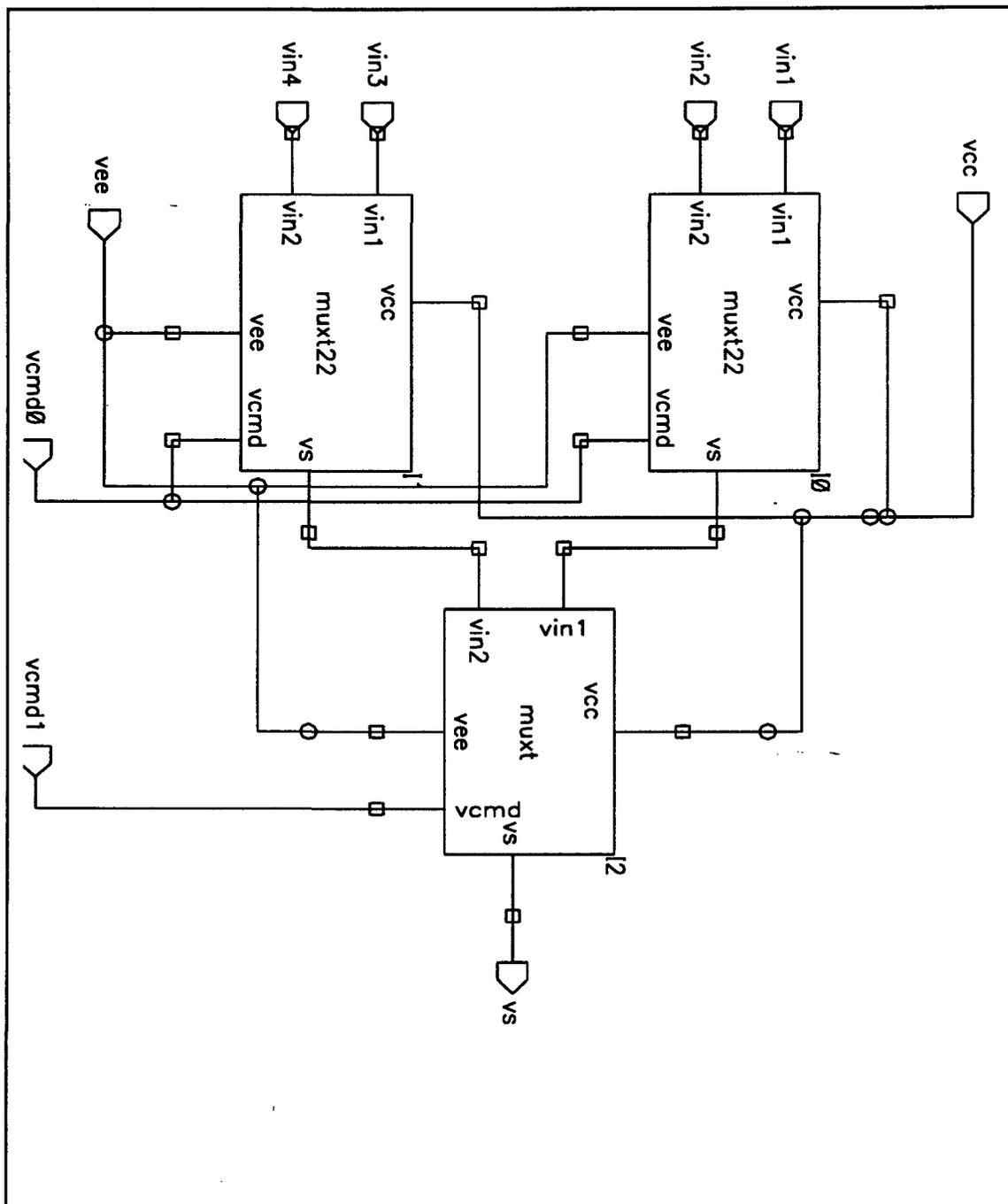


Figura 6.1 Matriz de Multiplexores usada en el diseño.

observar que hemos diferenciado entre los multiplexores llamando a unos MUXT y a otros MUXT22.

La diferencia entre los muxt22 y los muxt es que los muxt22 tienen un nivel de

continua negativo a la salida y que los muxt lo tienen positivo.

Tras la elección de la señal de entrada nos aseguramos de que la siguiente etapa no cargue a ésta utilizando un buffer de salida diseñado para tener una alta impedancia de

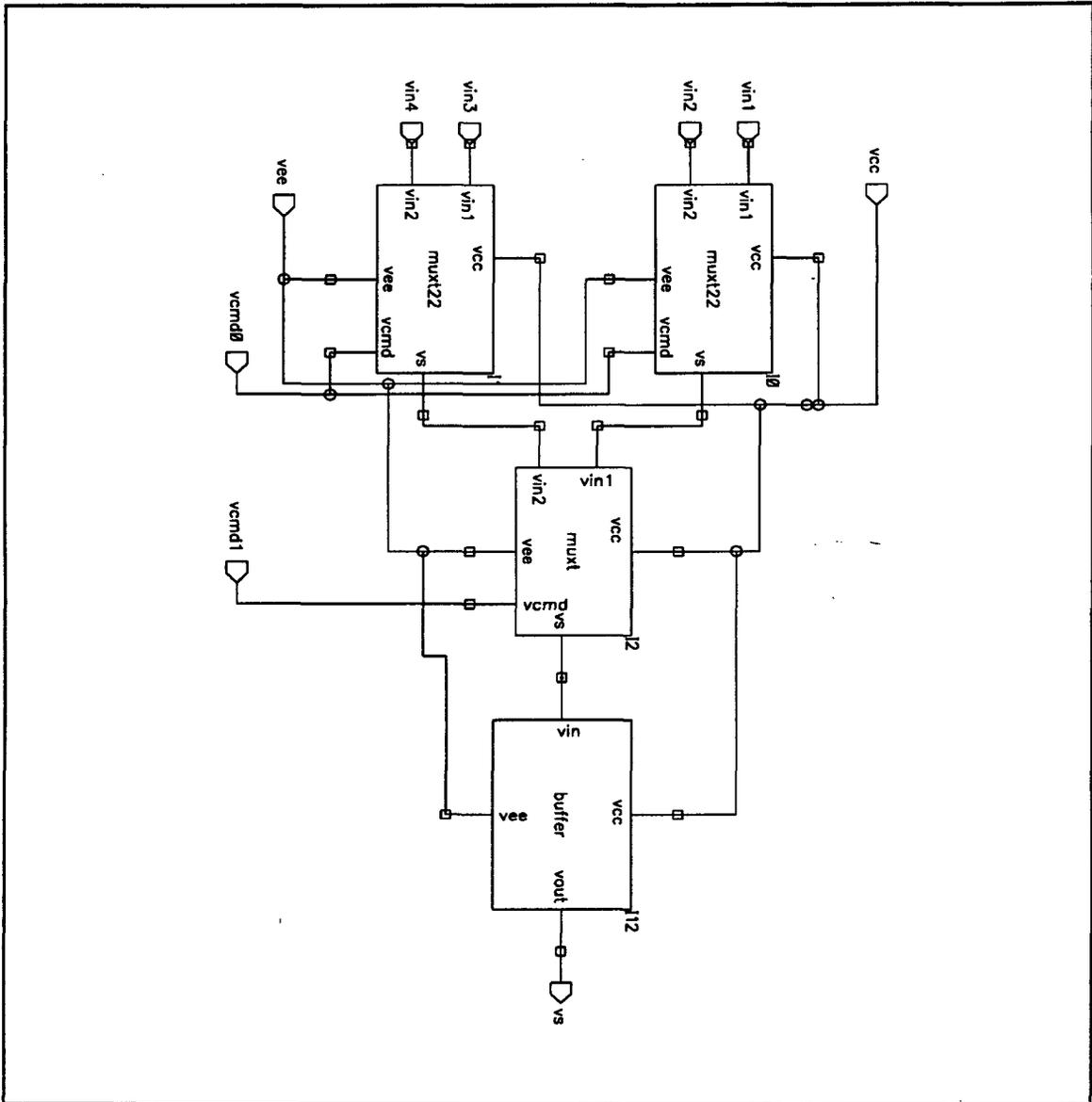


Figura 6.2 Diagrama de bloques de la etapa de Multiplexación y Adaptación.

entrada y baja impedancia de salida. Con esta célula la señal tendrá mayor potencia para atacar a las etapas posteriores.

En la figura 6.2 observamos el diagrama de bloques del diseño parcial (Multiplexación y adaptación).

El siguiente paso que se desea dar es el de poder controlar la ganancia de la señal elegida a través de una señal que nosotros introducimos. Para ello hemos utilizado el multiplicador visto en el capítulo anterior. Como ya se había nombrado para el funcionamiento que queremos dar nos hace falta que la señal llegue con un nivel de continua de unos tres voltios a la entrada de la célula. Para ello hemos dispuesto una etapa que manteniendo la señal igual la desplaza en nivel.

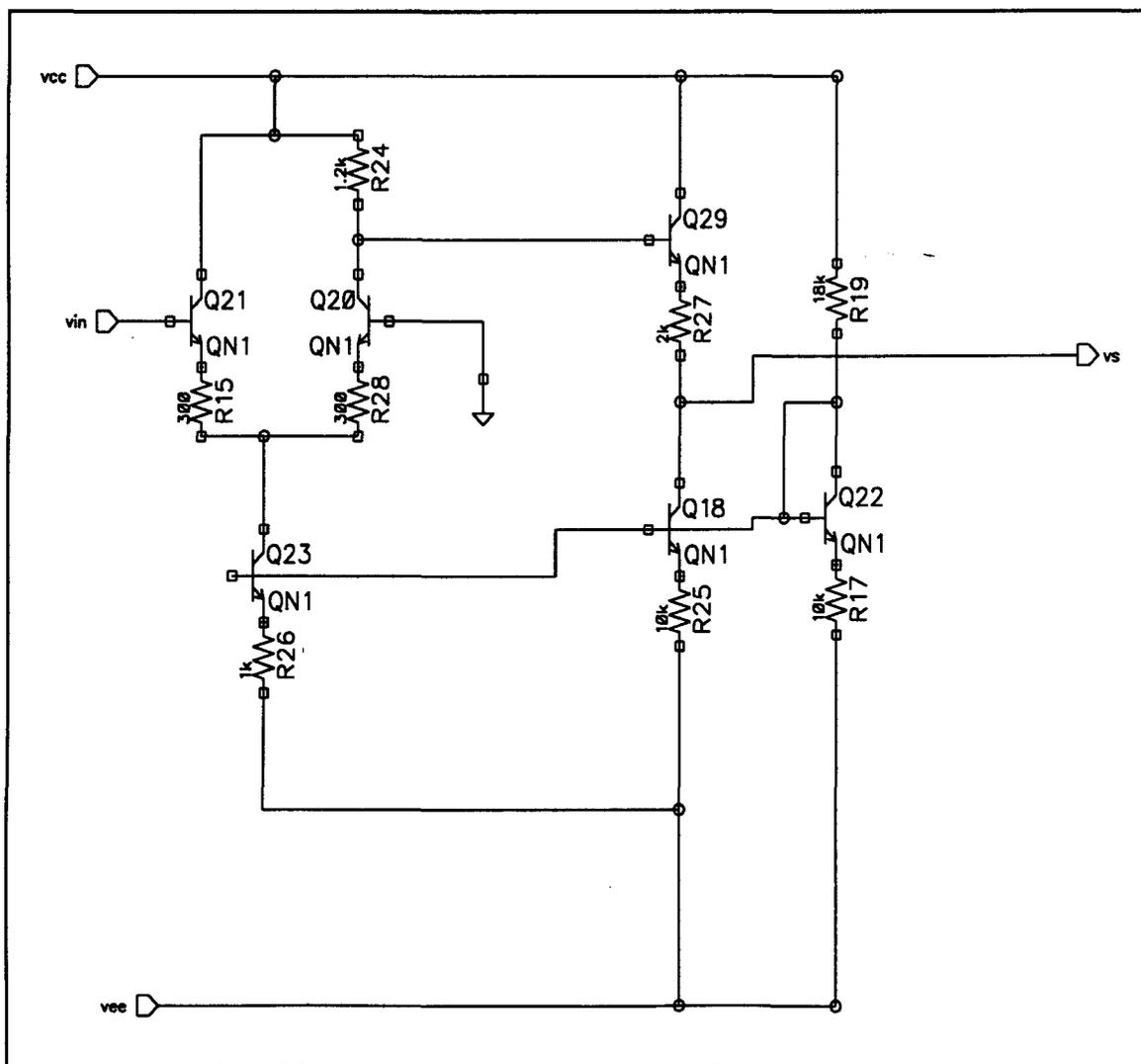


Figura 6.3.- Etapa de desplazamiento de nivel.



Una vez adaptado el nivel de continua a nuestras necesidades ya podemos conectar el control de ganancia. Para el uso que vamos a realizar de él necesitamos tener una señal de entrada y una de control. La señal de control será la que nos proporcione el nivel de salida deseado. A mayor nivel de la señal de control mayor será la ganancia (dentro de unos límites). Para demostrar su funcionamiento las simulaciones se han realizado utilizando una señal de control en forma de rampa, de tal forma que variaremos la ganancia de cero hacia arriba y luego volverá a disminuir hasta convertirse de nuevo en cero.

En la figura 6.5 se puede observar el diseño con el control de ganancia incluido. Este diseño no funcionará correctamente debido a que el nivel de continua con que llega la señal de entrada debe reflejarse en la segunda entrada del par diferencial para que la respuesta de salida sea simétrica. Para lograr una exactitud total en el circuito se ha optado por incluir una circuitería que refleje el camino seguido por la señal hasta ahora, obteniendo así el mismo nivel de continua que en la primera etapa.

Por último solo nos queda una etapa de amplificación de ganancia programable que tomará la salida balanceada del control de ganancia y nos dará la señal diferencial de salida. La ganancia de esta etapa la podremos variar mediante el cortocircuito de las resistencias de emisor al cual tendremos acceso mediante dos patillas del integrado que se acondicionarían para esta función.

Tenemos la posibilidad de hacer que la señal de salida sea asimétrica utilizando un convertidor diferencial asimétrico y conectándole un buffer a la salida. Aunque esta posibilidad queda contemplada no se ha realizado debido a la falta de espacio que tenemos en el layout disponible para diseñar en esta tecnología.

El esquema de lo realizado hasta ahora se muestra en la figura 6.7. En ella se tiene el

### Matriz de Conmutación con Control de Ganancia

esquema de una matriz de conmutación de cuatro entradas, con salida diferencial y control de ganancia externo.

El diseño, realizado totalmente con transistores bipolares, consta de 151 transistores bipolares y 250 resistencias obteniendo así un aprovechamiento de más del 50% de los elementos disponibles, lo cual es destacable en una metodología de *arrays*.

La realización del diseño físico ha sido realizada manualmente, respetando las reglas de diseño del fabricante. Se ha utilizado el modelo L1200 que tiene una superficie de 4,47\*2,86 (mm), 28 pads de salida y opera con una alimentación de 15 Voltios.

El diseño definitivo consta de 4 entradas, alimentación positiva y negativa, masas, 2 señales de selección, control de ganancia, salidas y la entrada y salida de un buffer.

Con estas características se ha elegido el encapsulado tipo DILC28 que dispone de 28 patillas (igual que el L1200) y tiene un área de 6,3\*6,3 (mm). En el apéndice II se muestran los tipos de encapsulados de los que se dispone en esta tecnología según las necesidades.

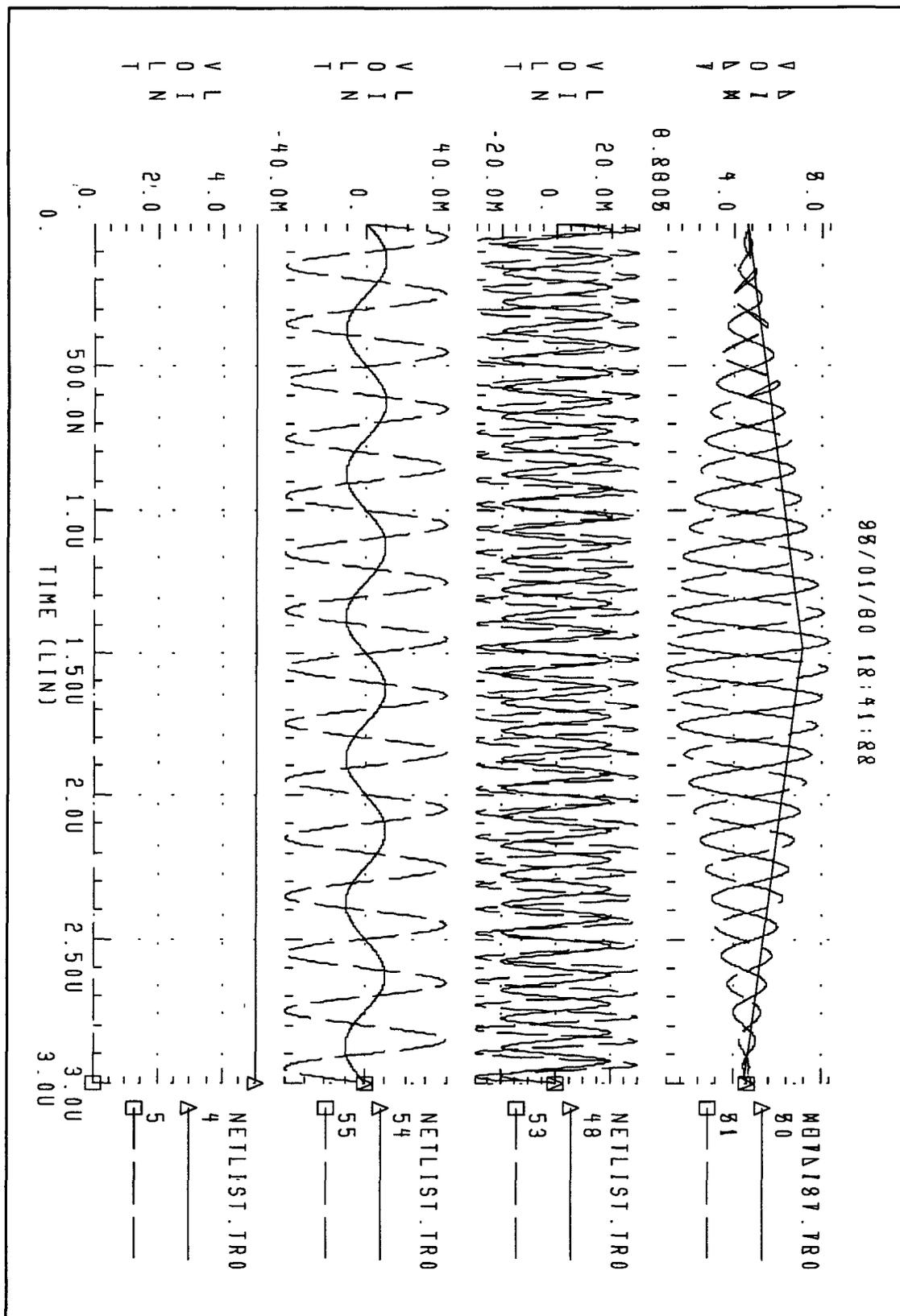


Figura 6.7.- Análisis Transitorio de la Matriz.

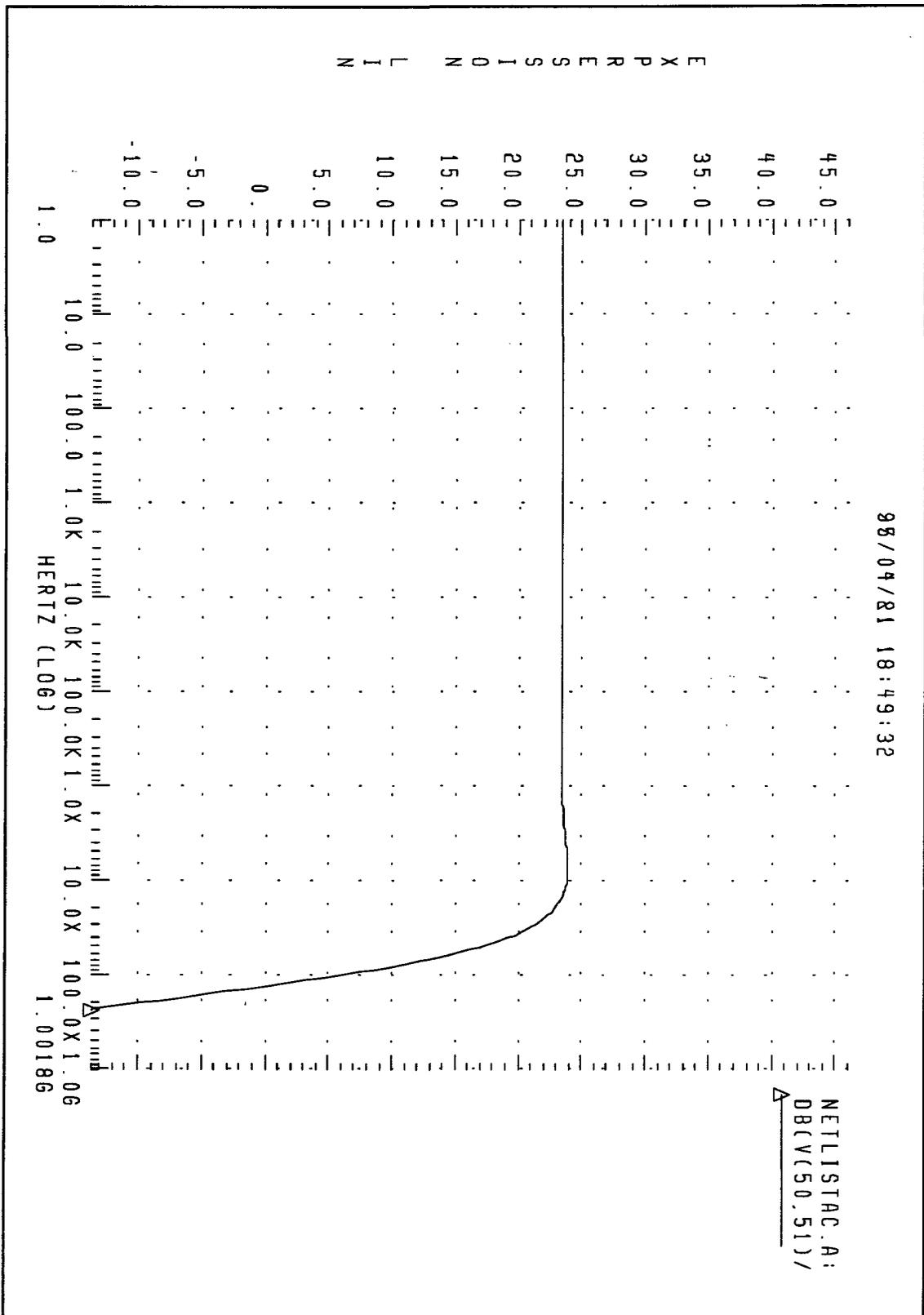


Figura 6.8.- Respuesta en Frecuencia.

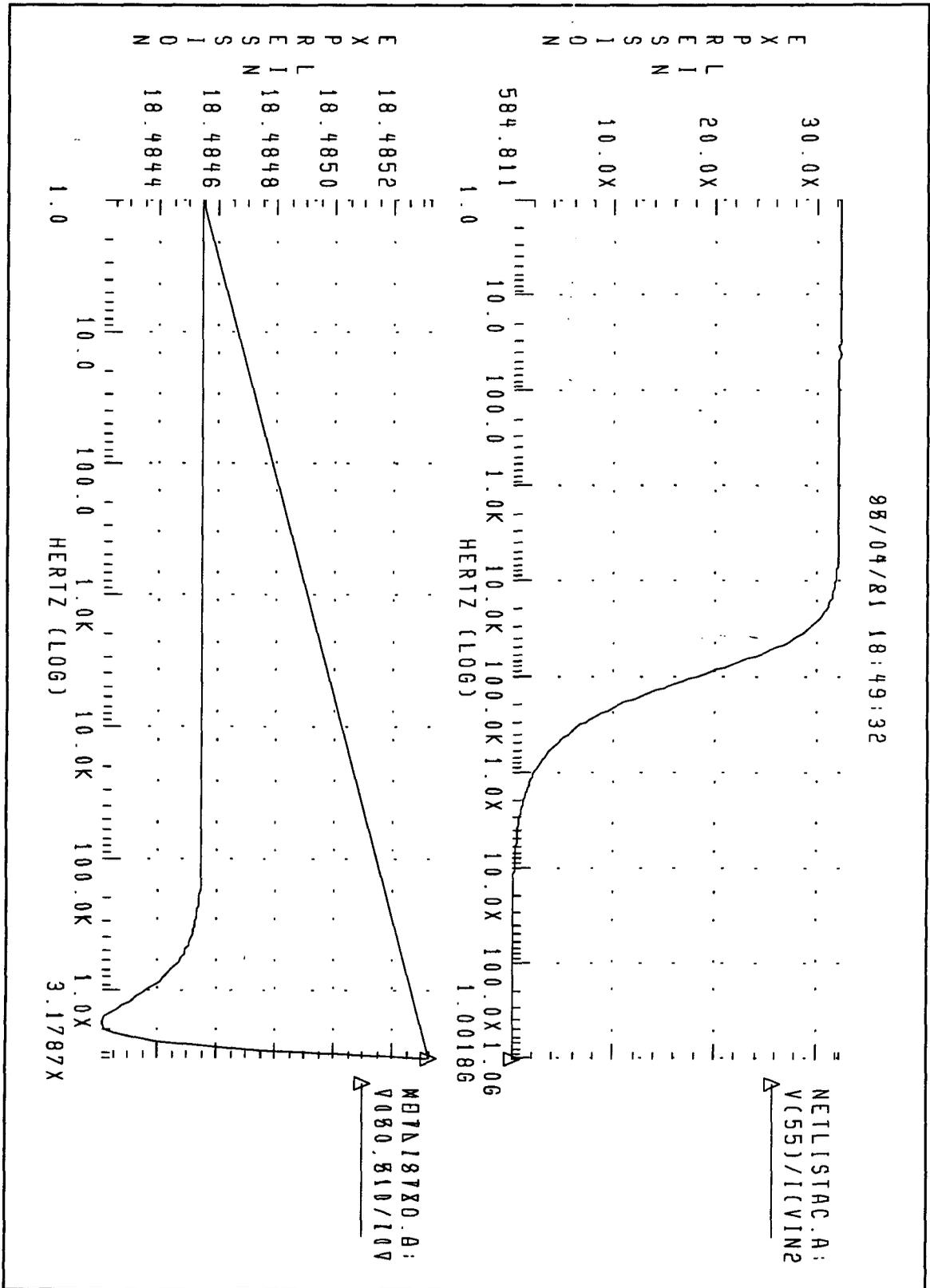


Figura 6.9.- Impedancias de entrada y salida.

© Del documento, los autores. Digitalización realizada por ULPGC. Biblioteca Universitaria, 2008

## **Características de la Matriz de Conmutación con Control de Ganancia**

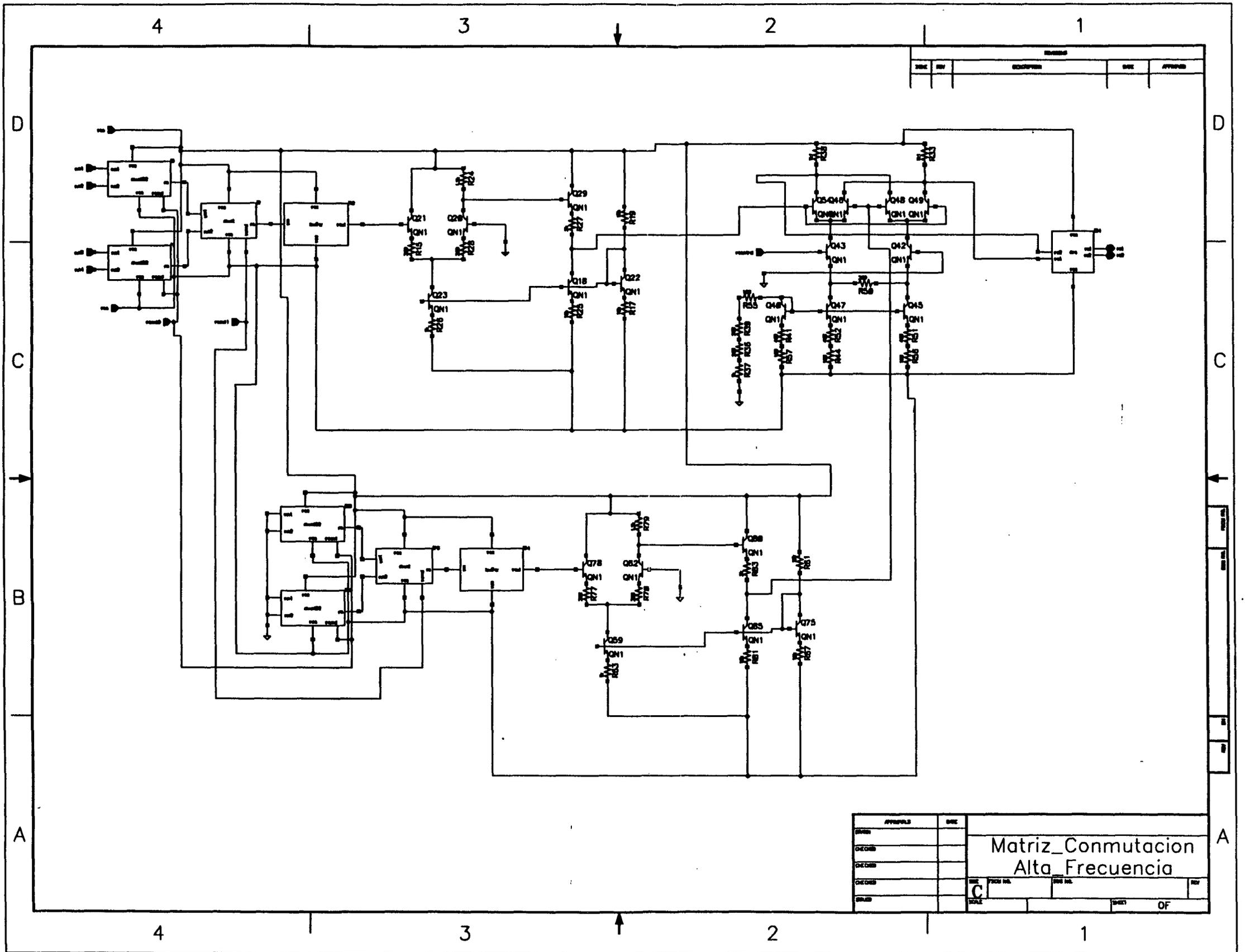
	<b>Matriz de Conmutación</b>
<b>Nº de Transistores</b>	<b>151</b>
<b>Nº de Resistencias</b>	<b>250</b>
<b>Área Utilizada</b>	<b>21 tiles</b>
<b>Alimentación</b>	<b>+7,5 / -7,5 Voltios</b>
<b>Ganancia en dB</b>	<b>Variable</b>
<b>Offset</b>	<b>0 Voltios</b>
<b>Ancho de Banda</b>	<b>35 MHz</b>
<b>Impedancia de Entrada (<math>Z_i</math>)</b>	<b>32 Kohmios</b>
<b>Impedancia de Salida (<math>Z_o</math>)</b>	<b>18 Ohmios</b>
<b>Potencia</b>	<b>1,2 Watios</b>

Tecnología: Bipolar Polyuse L de Thomson

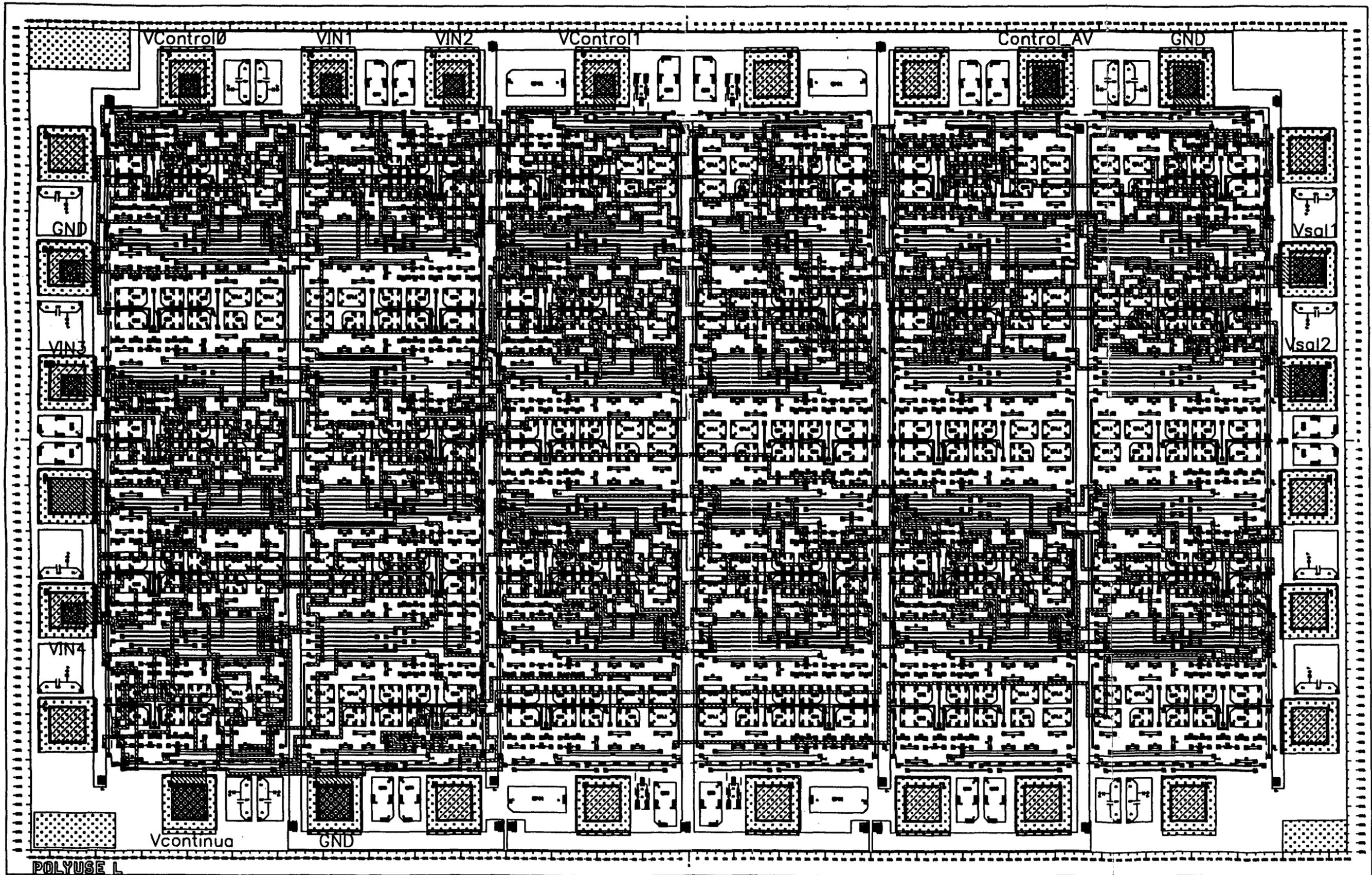
Simulaciones: Hspice

Fecha: Enero 1995

Observaciones: Diversas Configuraciones



MATRIZ\_4\_a\_1\_CON\_CONTROL\_AV\_PARA\_ALTA\_FRECUENCIA



# Capítulo VII

## Pliego de Condiciones

El documento de Pliego de Condiciones establece de manera definitiva las condiciones de realización de este proyecto.

Todas las cláusulas señaladas en este pliego de condiciones habrán de ser respetadas con obligado cumplimiento de las mismas.

Debido a que el proceso de fabricación del diseño lo realizará una empresa dedicada y dueña de la tecnología utilizada se delegará el Pliego de condiciones a aquel establecido por esta empresa para la fabricación de los circuitos.

Cualquier variación a realizar sobre el diseño, por pequeña que sea, deberá ser comunicada por escrito y documentada al autor del diseño para la aceptación de estas modificaciones. Serán admitidas todas aquellas modificaciones que se indiquen de manera clara en forma escrita con indicación expresa del coste adicional.

# Capítulo VIII

## Conclusiones

El tiempo que se ha dedicado al estudio de la tecnología bipolar Polyuse L de las series TSFL de Thomson nos ha llevado a obtener una serie de conclusiones que a continuación se desglosan.

La realización del diseño sin tener antecedentes ninguno utilizando unos 150 transistores bipolares ha tenido bastantes complicaciones, pero una vez encontrada una ruta de diseño bastante fiable se puede decir que el diseño con esta tecnología puede ser rápido según se obtenga experiencia en la realización del diseño físico. Sin embargo es de destacar que el no poder realizar ruteados automáticos es el mayor problema de esta tecnología.

En cuanto a la utilización de tecnología bipolar llegamos a la conclusión de que si lo que se busca es bajo nivel de ruido, ancho de banda y ganancia la tecnología bipolar puede ser una solución óptima. Inconvenientes de ésta son el mayor área que ocupa y el alto consumo de corriente que posee.

La decisión de utilizar esta tecnología deberá ser tomada después de haber decidido a que tipo de característica se le va a dar prioridad y actuar en consecuencia.

En cuanto a los arrays de transistores destacar la ventaja de su menor coste con respecto a diseños full-custom y la practicidad del ruteado tras haber obtenido la experiencia suficiente.

Tambien se quiere destacar en este capítulo el inconveniente de la crisis en que, en estos momentos, está la empresa Thomson, ya que la informacion, la rapidez y avances se ven afectados por ella.

# Capítulo IX

## Presupuesto

Para la realización del presupuesto del diseño se han tenido en cuenta los siguientes aspectos: El precio del *software*, el uso de las máquinas, el coste del diseño y la mano de obra.

### 8.1.- *Software*.

Coste de compra.....	1500 ecus.
Amortización en 3 años , 2 proyectos realizados.....	250 ecus.
Total pesetas.....	40.000 ptas.

### 8.2.- Máquinas.

Amortización.....	1228 ptas/hora.
Actualización.....	1228 ptas/hora.
Total uso de la máquina.....	2456 ptas/hora.
Total .....	100 horas*2456ptas/hora=245600 ptas.

### 8.3.- Coste fabricación.

Precio fabricación.....	500 ecus/mm <sup>2</sup>
-------------------------	--------------------------

Total.....16\*500=8000 ecus=1280000 ptas.

**8.4.- Mano de Obra.**

Hora de trabajo.....3500 ptas

Total horas.....180 horas.

Total.....180\*3500= 630000 ptas

**TOTAL: 40000+245600+630000+1280000= 2195600 ptas.**

# Apéndices

## **Apéndice I**

En este Apéndice se ha incluido la información suministrada por el fabricante de los distintos tipos de encapsulados disponibles para ubicar los diseños.

En él también se incluyen una serie de reglas de conexión entre los pads y las patillas del encapsulado para evitar problemas bastante comunes.

## 1. Introduction

This chapter provides the customer with the necessary information about each package available in the ASIC product line, with relation to the TSFL series.

We have pre-selected the packages available for each array with respect to cavity, die size and wire length compatibility with TMS packaging rules. Nevertheless a description of the features of each package is supplied. This enables the designer of printed board and integrated circuits:

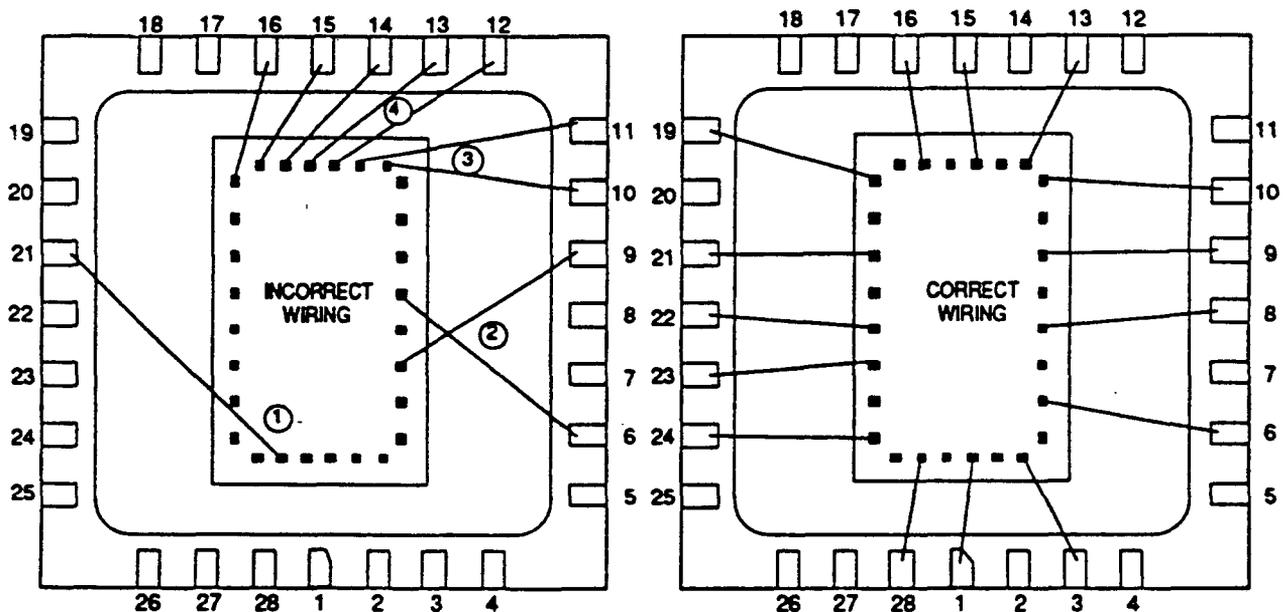
- to keep the external dimensions of packages in line with tolerances,
- to access package drawings and view the IC pin-out,
- to adapt his package to power dissipation requirements, thanks to the thermal resistance information supplied for each package.

For further information please consult the TMS packaging manual. We strongly advise you to follow the supplied bonding recommendations when defining your application's pin-out.

## 2. Bonding recommendations

When selecting the pin-out, some assembly limitations must be taken into consideration.

- 1 The pads used should be regularly positioned around the chip so as to minimize and obtain equal wire length. Examples of correct and incorrect pin-outs are illustrated below.
- 2 Chip axis must be aligned with the package axis so as to avoid long wires and allow minimum space between the chip and cavity. In the case of packages with less than 24 pins, the cavity is of rectangular shape and the largest side of the chip must be parallel to the largest side of the cavity.
- 3 Bonding wires cannot cross each other.
- 4 Bonding wires cannot cross the active area.
- 5 For all packages, pin 1 is located close to the center of one cavity side.



Under any circumstance, you are strongly recommended to contact your design support center before selecting the package type.

### 3. Matching power dissipation requirement and packaging:

**Power dissipation calculations:**

The basis procedure that a designer might use to calculate power dissipation for an ASIC is given in the related User's manual. Ask your local Sales Office for a copy.

**Thermal resistance of packages:**

Combined with power dissipation, the thermal resistance of a package has to be taken into account for the final choice of the package. This will have effects on reliability figures.

Considering:

- TA as the operating room temperature,
- PT as the total power dissipated,
- RJAT as chip-ambient thermal resistance,

The junction temperature is to be calculated using the following formula:

$$T_J = T_A + R_{JAT} \times P_T$$

T<sub>Jmax</sub>=150 °C

### 4. Available standard packages for POLYUSE L arrays

Array	DIL							LCCC			LDCC		PLCC	
	16	18	20	24	28	40	48	20	28	44	28	44	28	44
TSFL 06	C/P	C/P	C/P					C			C		P	
TSFL 12	C	C/P	C/P	C/P	C				C		C		P	
TSFL 19				C/P	C/P	C/P	C		C	C	C	C	P	P

C = ceramic

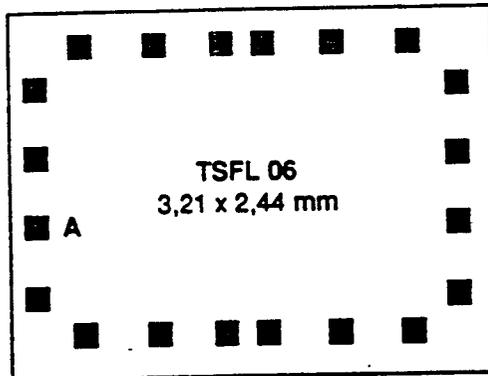
P = plastic

- Non standard packaging solutions can be examined on special request.
- Small Outline plastic packages also available on request.

## 5. TSFL 06 CHIP



SCALE : 15



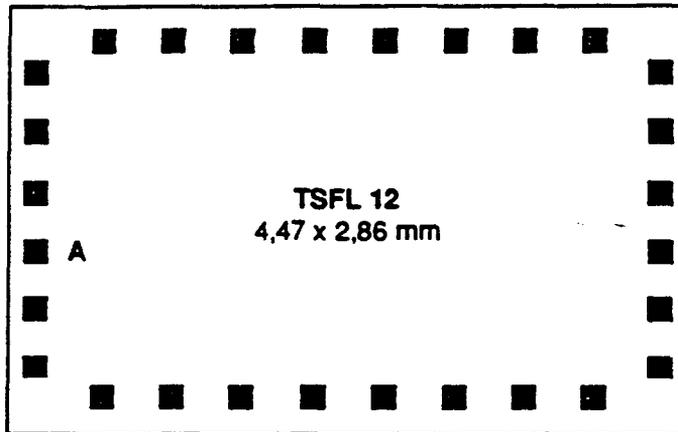
SCALE : 20

*note the different scales*

## 6. TSFL 12 CHIP



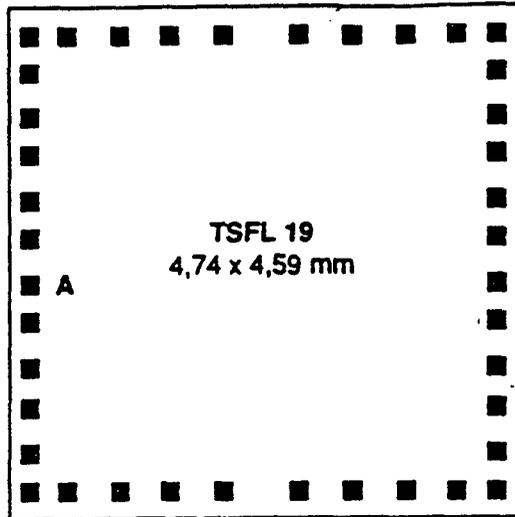
SCALE : 15



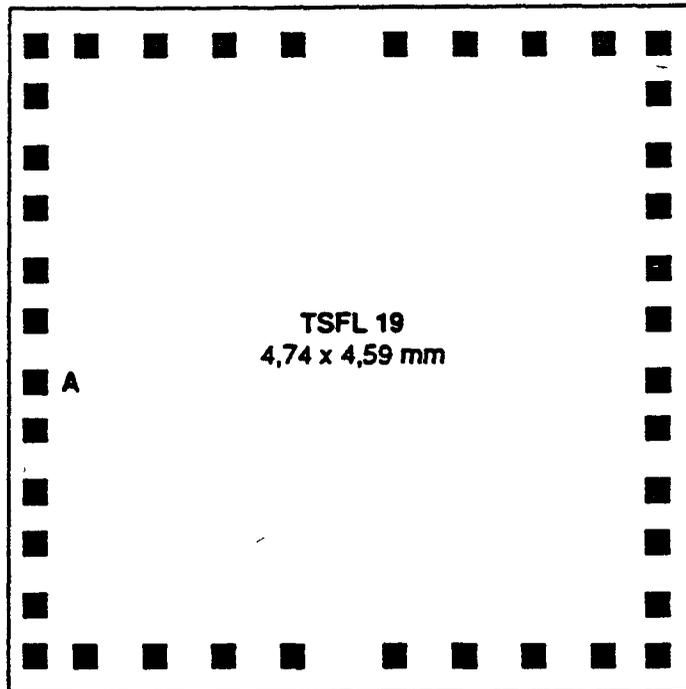
SCALE : 20

• *note the different scales*

## 7. TSFL 19 CHIP



SCALE : 15



SCALE : 20

\* note the different scales

## **8. Packaging**

In the following pages we include several examples of packages in order to show the usual connection layout round the cavity. However for the same PIN-OUT the layout may differ slightly from one supplier to another.

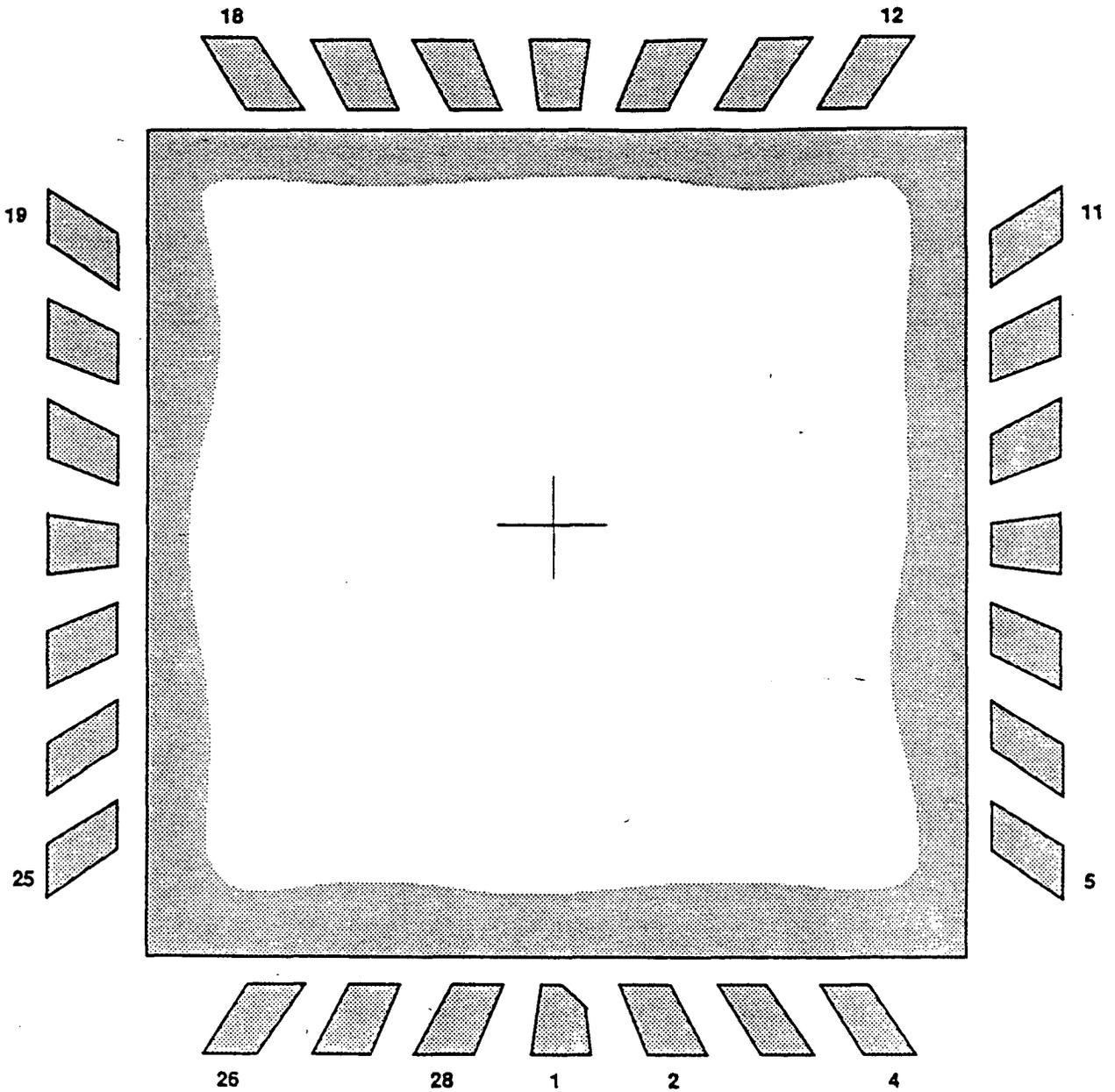
These examples should enable you to choose the layout for the chip's operative pads, in order to optimize the length of package connections.

If you do not find what you require amongst the examples, please consult our sales representative. We may then study other possible assembly layouts with you.

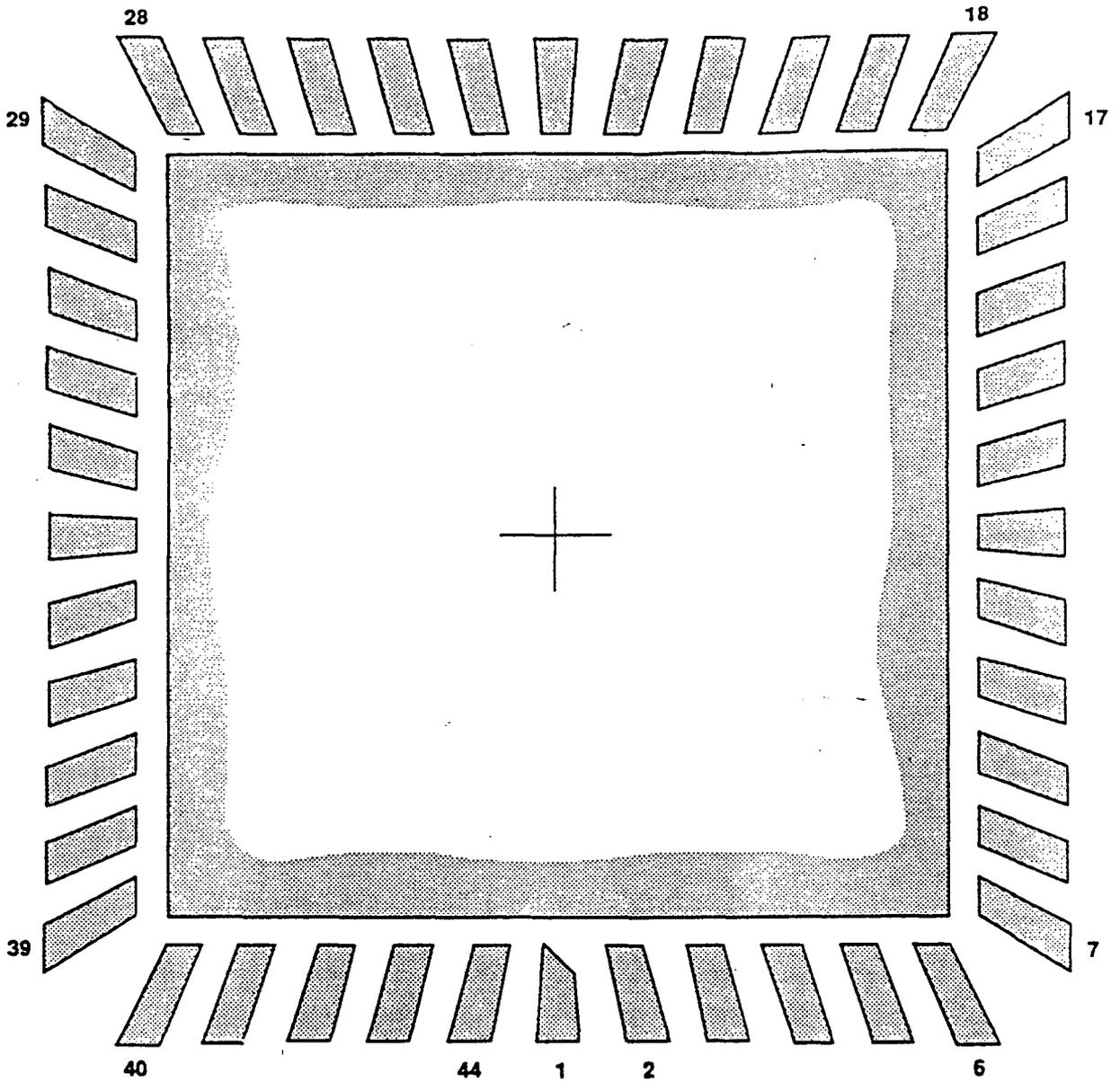
In all circumstances you should request his approval of your completed assembly layout.

## **9. How to draw the bonding diagram**

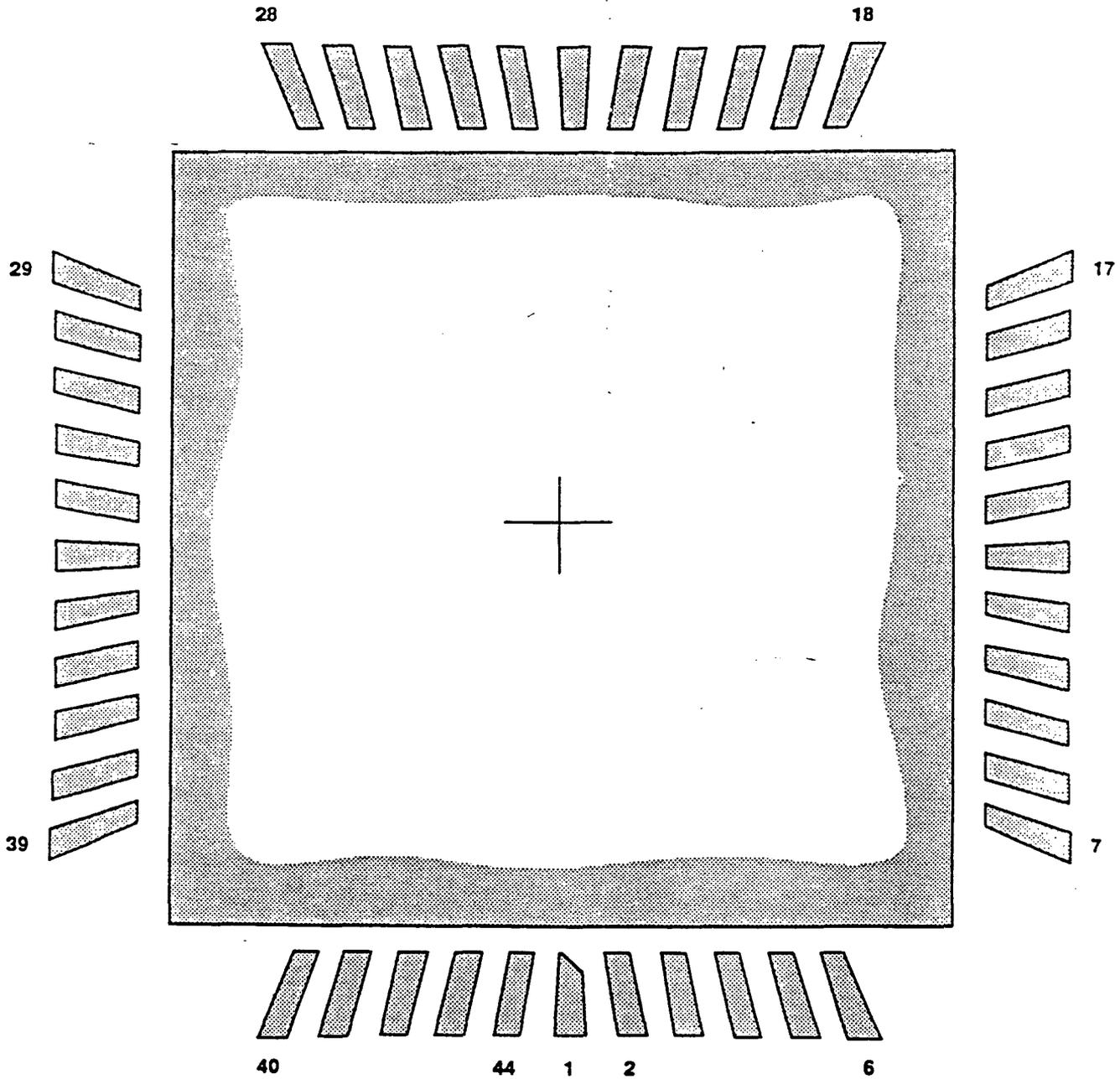
- Select the package.
- Photo copy the package frame.
- Photo copy the chip outline drawing.
- Select the die representation with the scale factor to fit the package frame scale factor.
- Cut the die plot photocopy and stick it on the package frame photocopy with the proper orientation.
- Draw the wires between the proper die pads and package pads.
- Place the originals back in this manual.



<p>TYPE <b>LCCC 28</b></p>	<p>CAVITY SIZE (mm) <b>6.3 x 6.3</b></p>	<p>SCALE <b>20</b></p>
--------------------------------	--	----------------------------

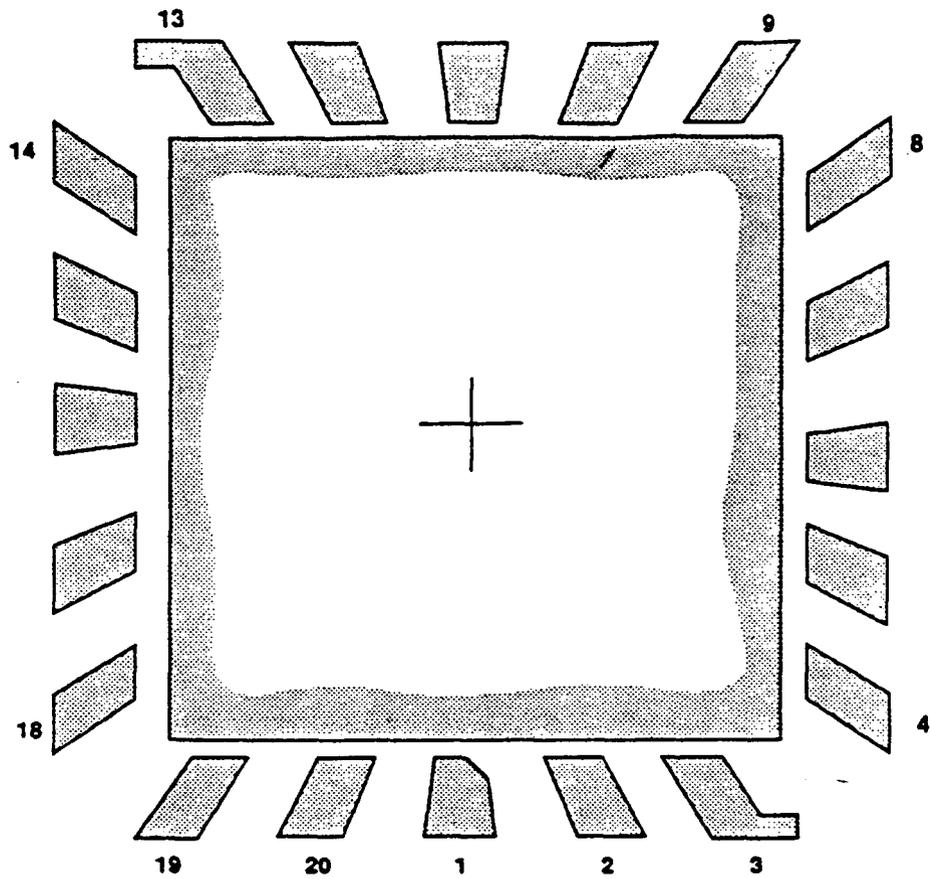


<p>TYPE <b>LCCC 44</b></p>	<p>CAVITY SIZE (mm) <b>7.6 x 7.6</b></p>	<p>SCALE <b>15</b></p>
--------------------------------	--	----------------------------



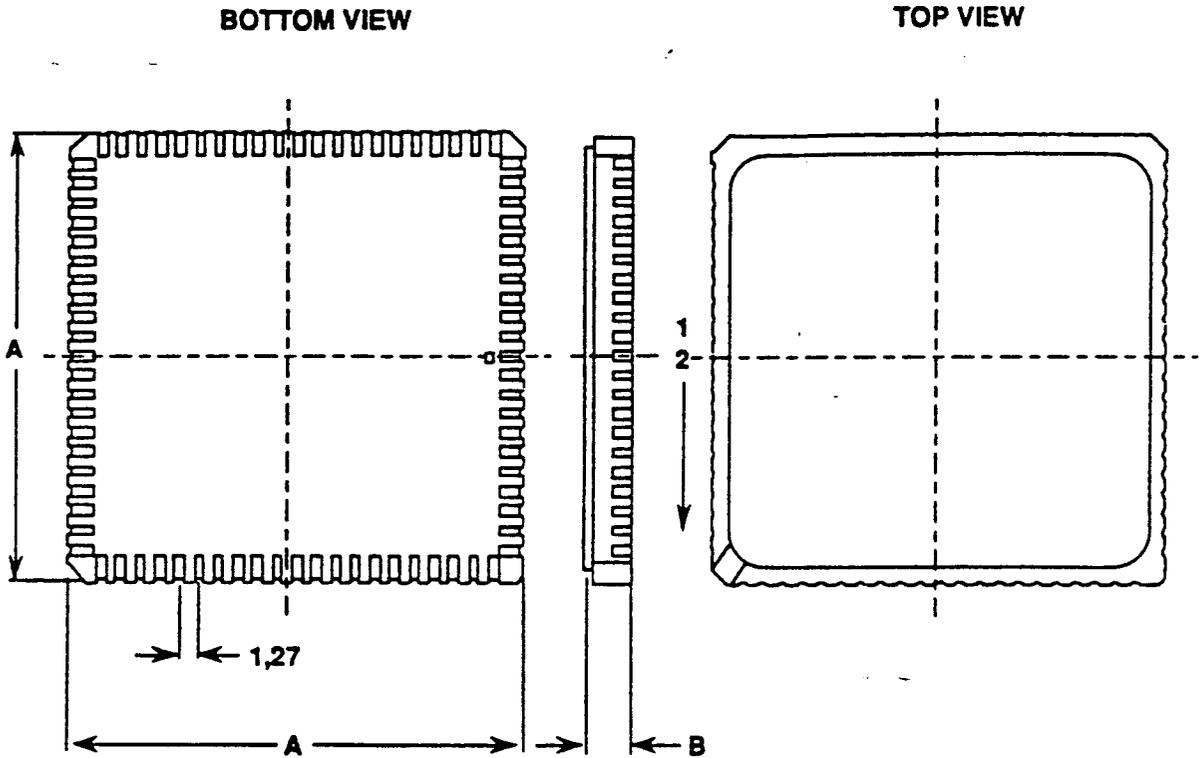
© Del documento, los autores. Digitalización realizada por ULPGC. Biblioteca Universitaria, 2006

<p>TYPE <b>LCCC 44</b></p>	<p>CAVITY SIZE (mm) <b>8.6 x 8.6</b></p>	<p>SCALE <b>15</b></p>
--------------------------------	--	----------------------------

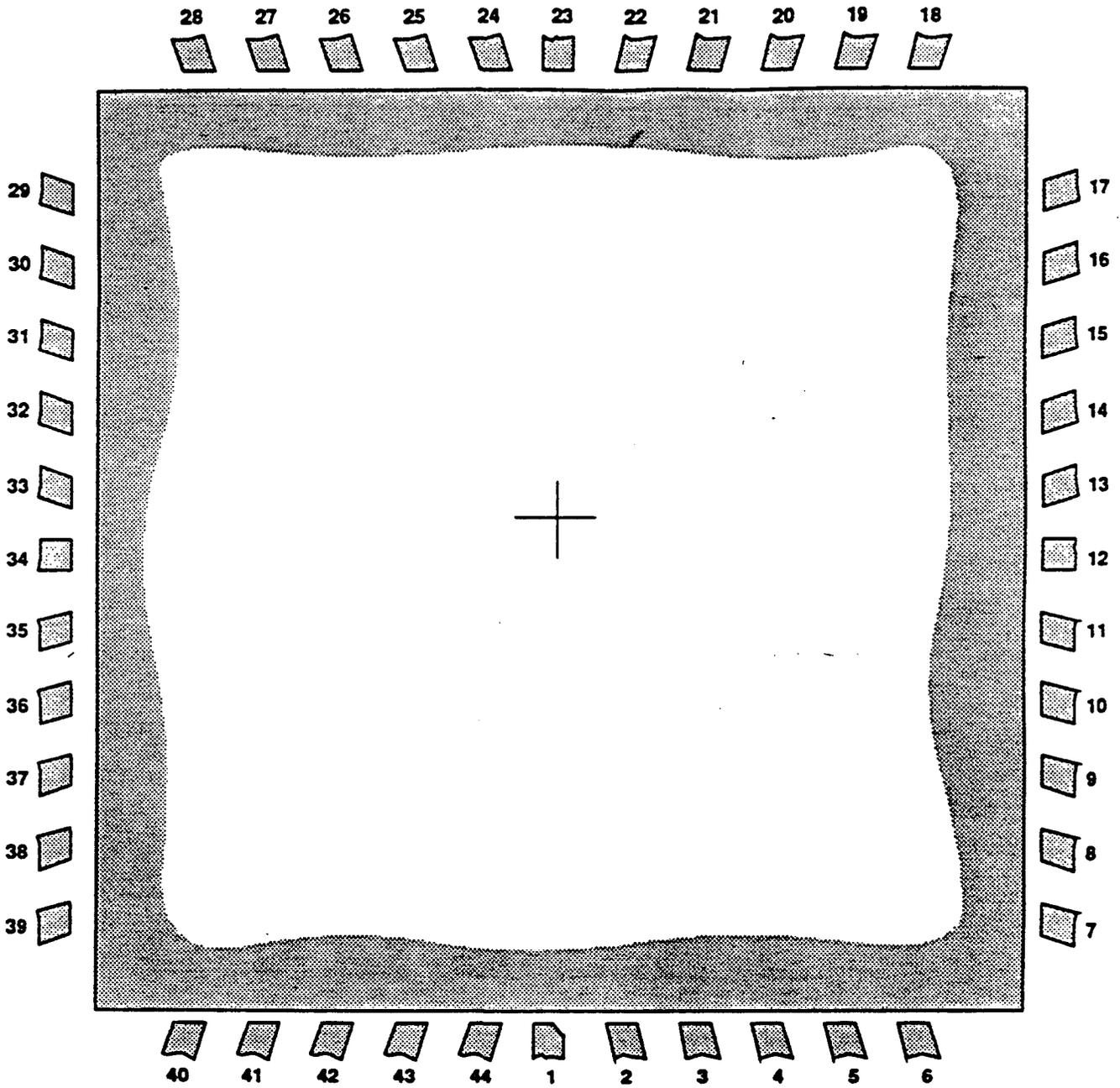


<p><b>TYPE</b> <b>LCCC 20</b></p>	<p><b>CAVITY SIZE (mm)</b> <b>4.1 x 4.1</b></p>	<p><b>SCALE</b> <b>20</b></p>
---------------------------------------	---	-----------------------------------

LCCC packages:



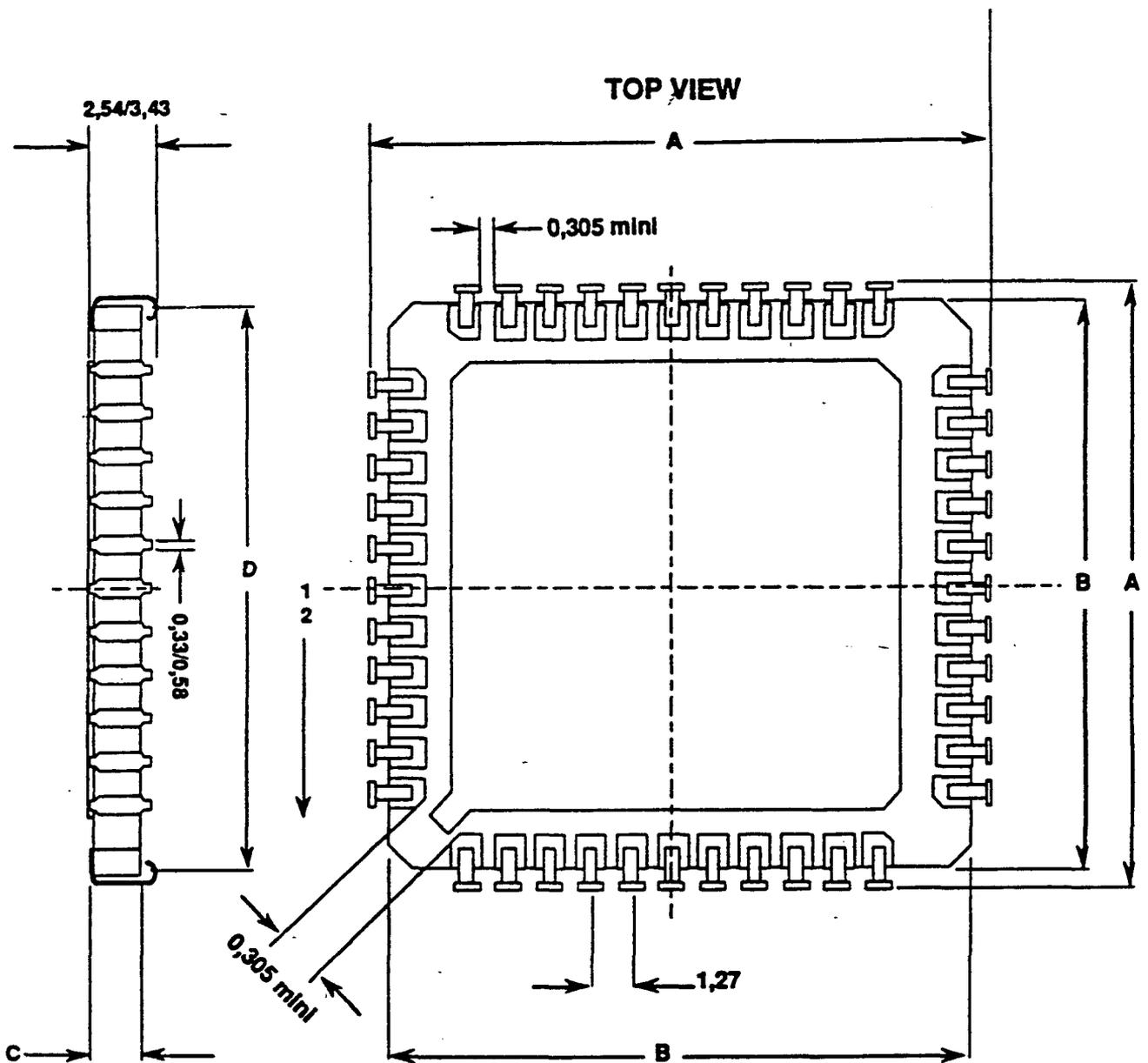
NO OF TERMINALS	A		B	
	MIN	MAX	MIN	MAX
20	8,69	9,09	1,52	2,54
28	11,23	11,68	1,52	2,54
44	16,26	16,81	1,63	3,05



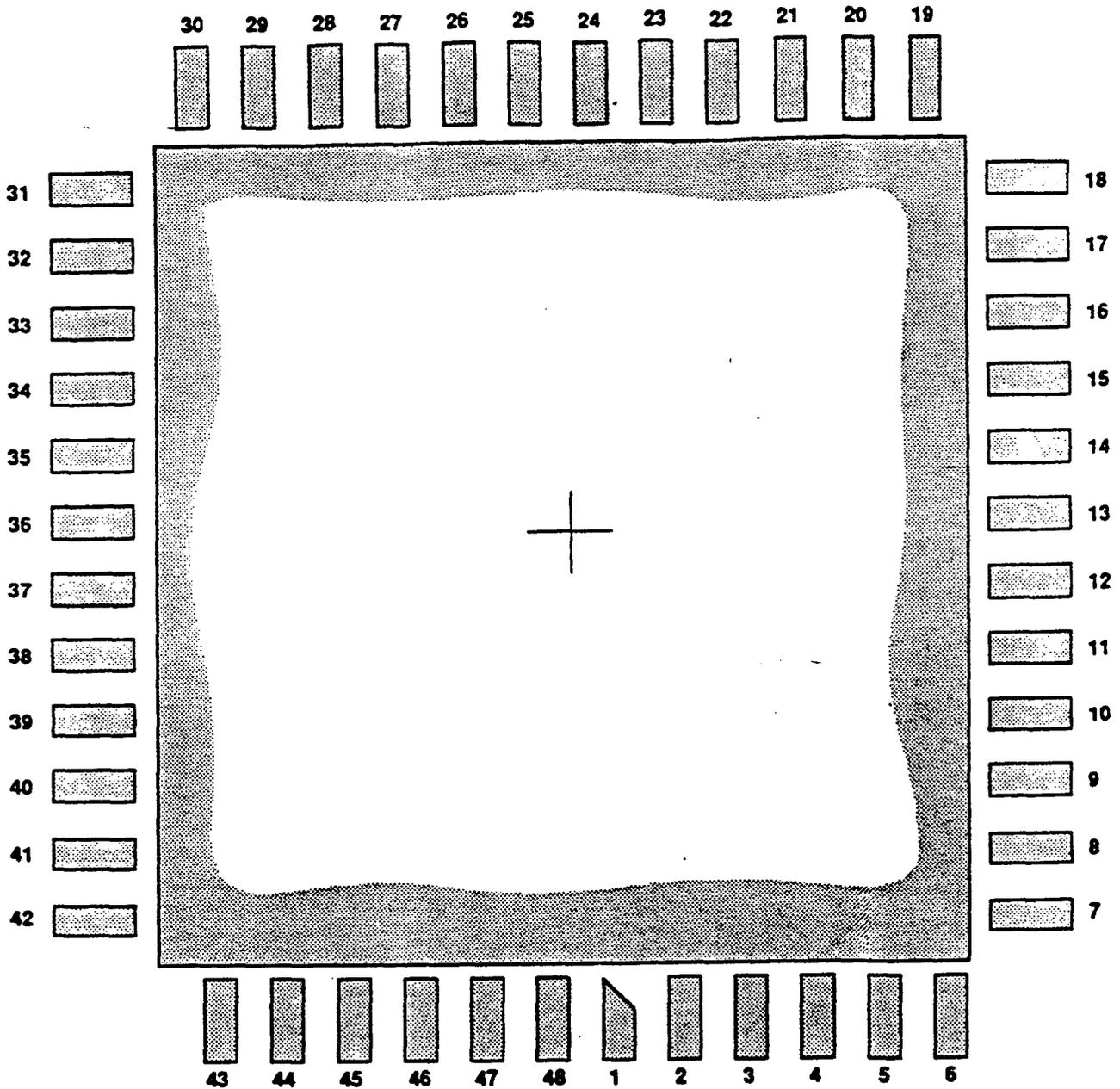
© Del documento, los autores. Digitalización realizada por ULPGC. Biblioteca Universitaria, 2006

<p>TYPE <b>LDCC 44</b></p>	<p>CAVITY SIZE (mm) <b>7.6 x 7.6</b></p>	<p>SCALE <b>20</b></p>
--------------------------------	--	----------------------------

LDCC packages

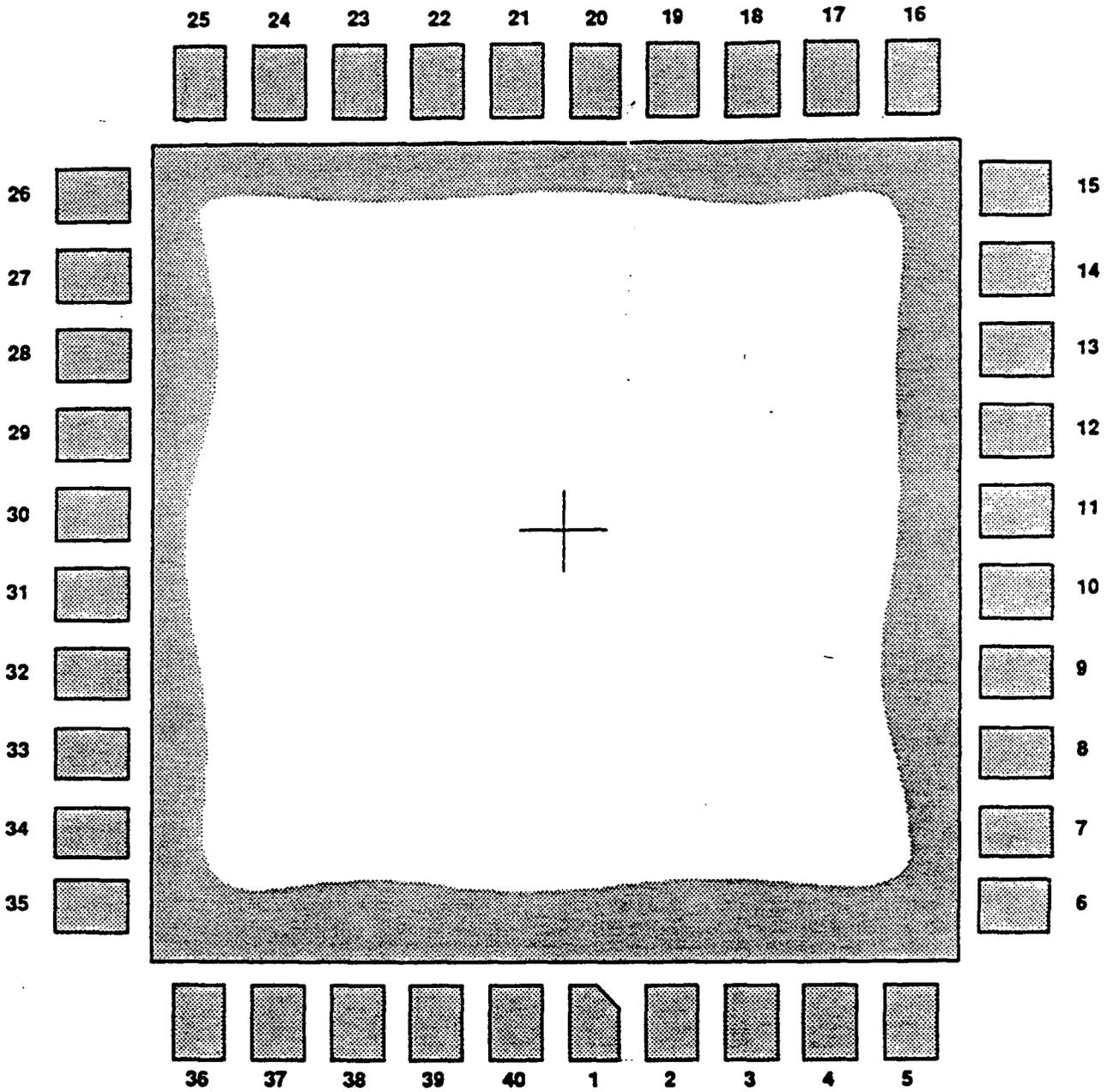


LEAD	A		B		C		D	
	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
28	-	12,55	11,23	11,63	1,37	1,98	-	-
44	17,27	17,78	15,95	16,81	1,37	1,98	15,49	16,51

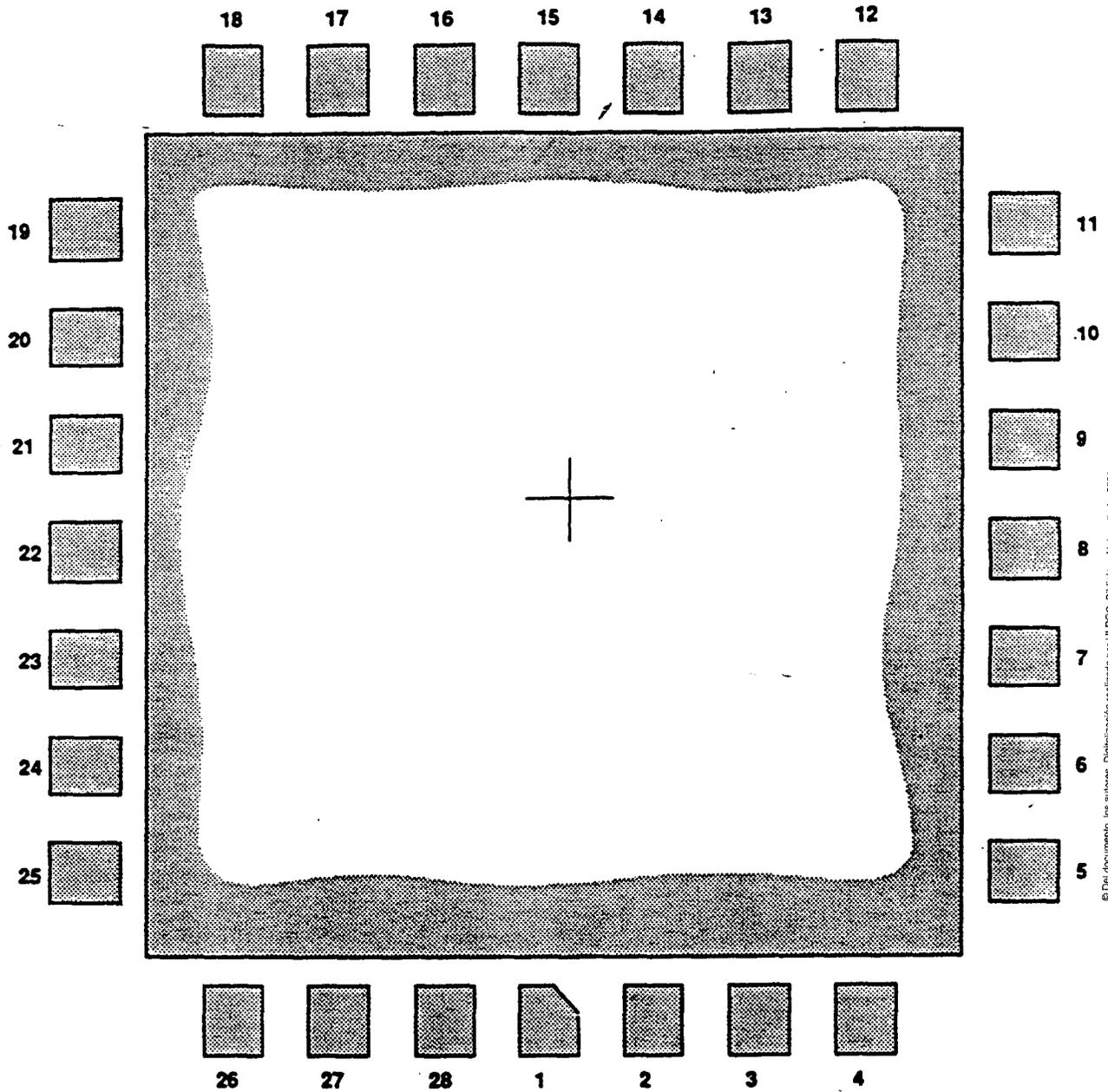


© Del documento, los autores. Digitalización realizada por ULPGC. Biblioteca Universitaria, 2008

<p>TYPE <b>DILC 48</b></p>	<p>CAVITY SIZE (mm) <b>6.3 x 6.3</b></p>	<p>SCALE <b>20</b></p>
--------------------------------	--	----------------------------

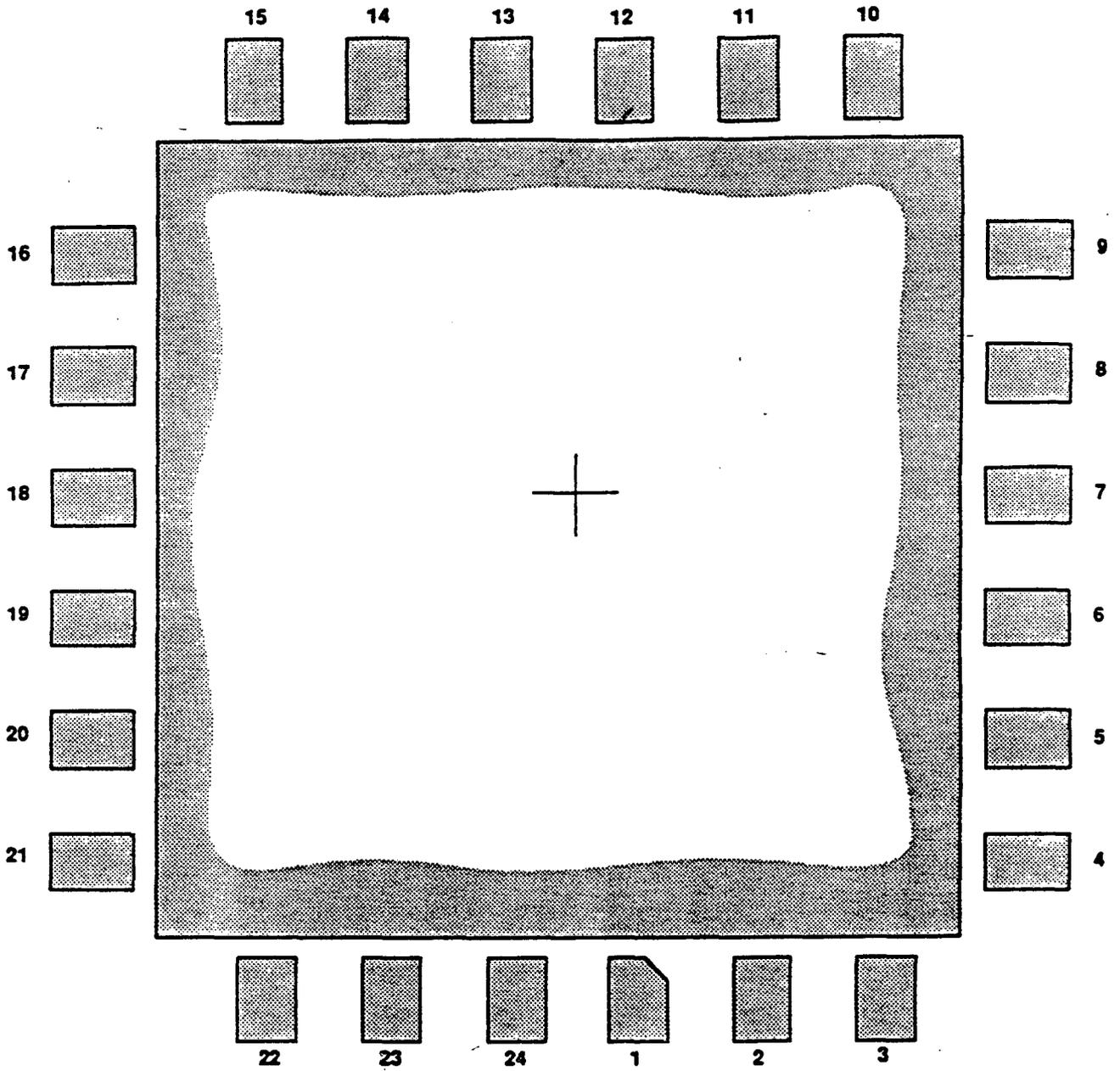


<p><b>TYPE</b> <b>DILC 40</b></p>	<p><b>CAVITY SIZE (mm)</b> <b>6.3 x 6.3</b></p>	<p><b>SCALE</b> <b>20</b></p>
---------------------------------------	---	-----------------------------------



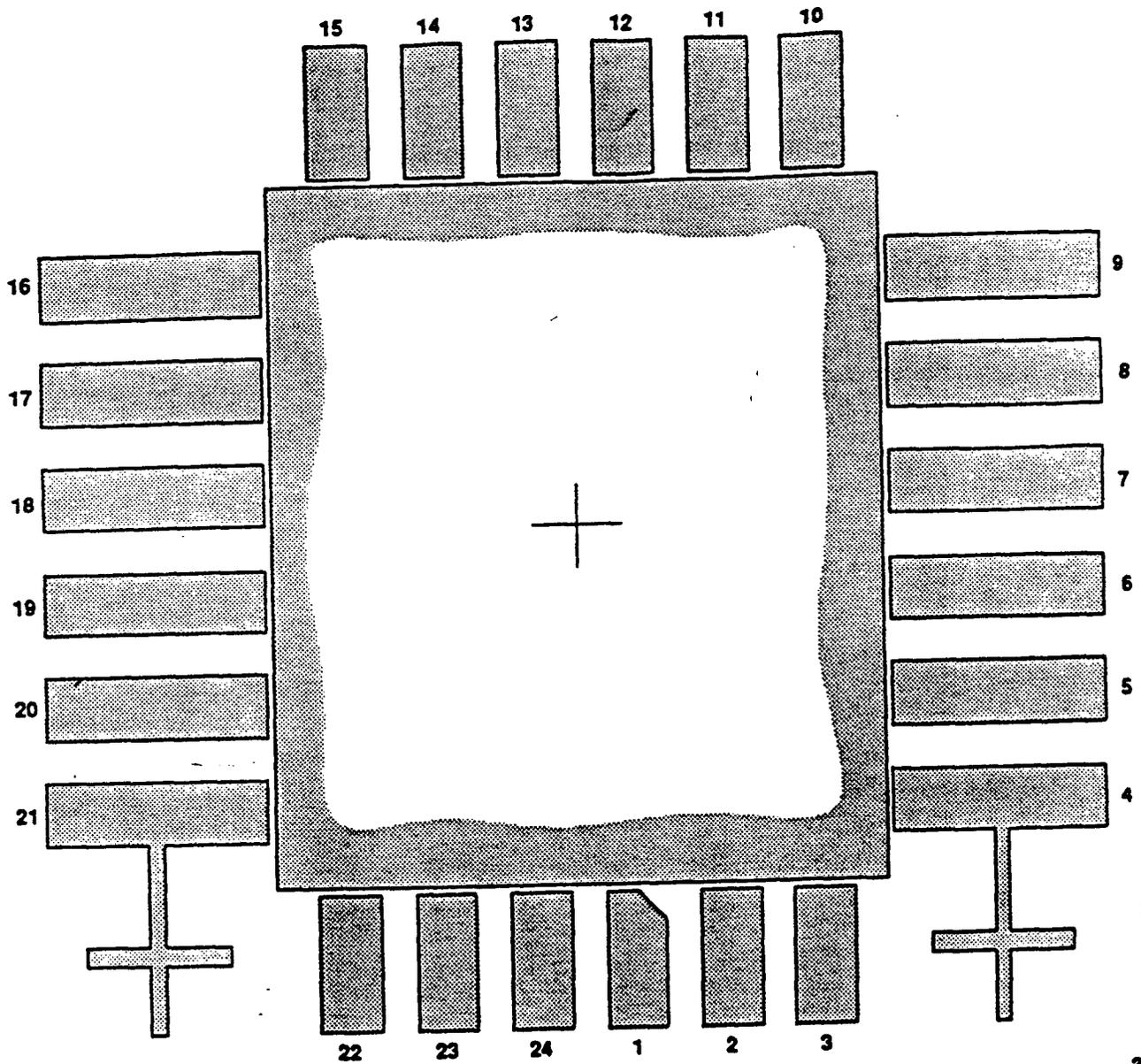
© Del documento, los autores. Digitalización realizada por ULPGC. Biblioteca Universitaria, 2008

<p style="text-align: center;"><b>TYPE</b> <b>DILC 28</b></p>	<p style="text-align: center;"><b>CAVITY SIZE (mm)</b> <b>6.3 X 6.3</b></p>	<p style="text-align: center;"><b>SCALE</b> <b>20</b></p>
---	---	---

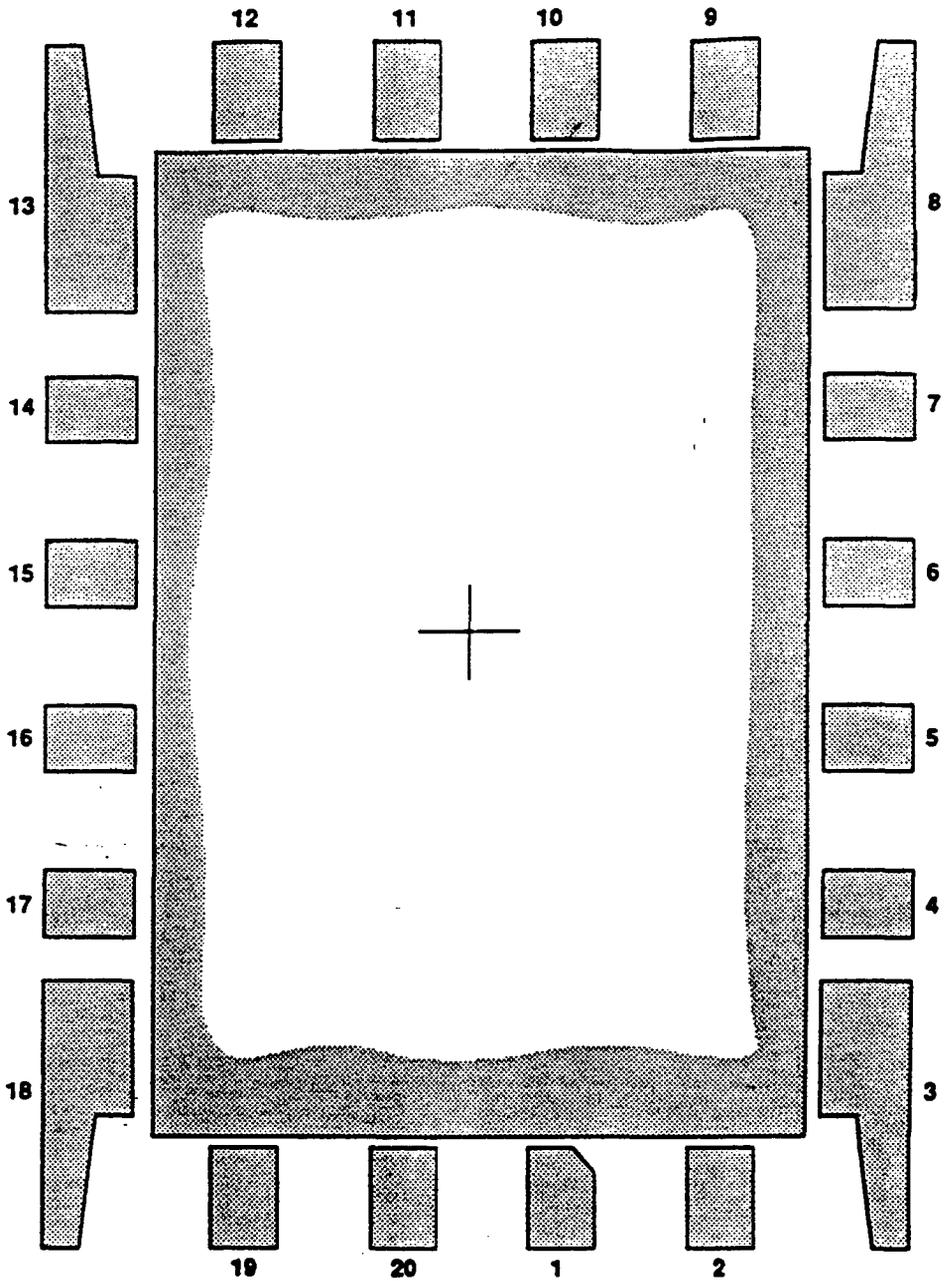


© Del documento: los autores. Digitalización realizada por ULPGC. Biblioteca Universitaria, 2008

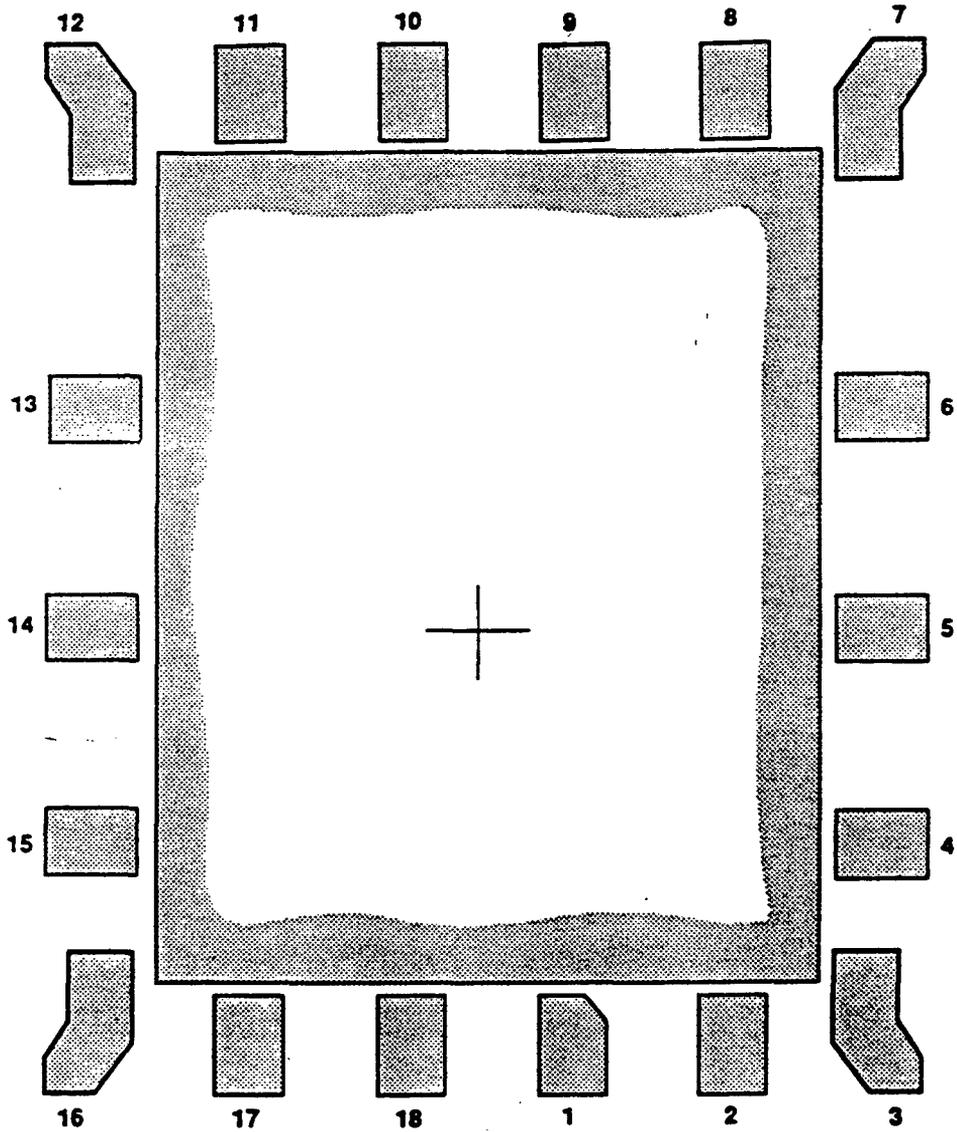
<p>TYPE <b>DILC 24</b></p>	<p>CAVITY SIZE (mm) <b>6.3 x 6.3</b></p>	<p>SCALE <b>20</b></p>
--------------------------------	--	----------------------------



<p>TYPE <b>DILC 24</b></p>	<p>CAVITY SIZE (mm) <b>5.3 x 4.6</b></p>	<p>SCALE <b>20</b></p>
--------------------------------	--	----------------------------

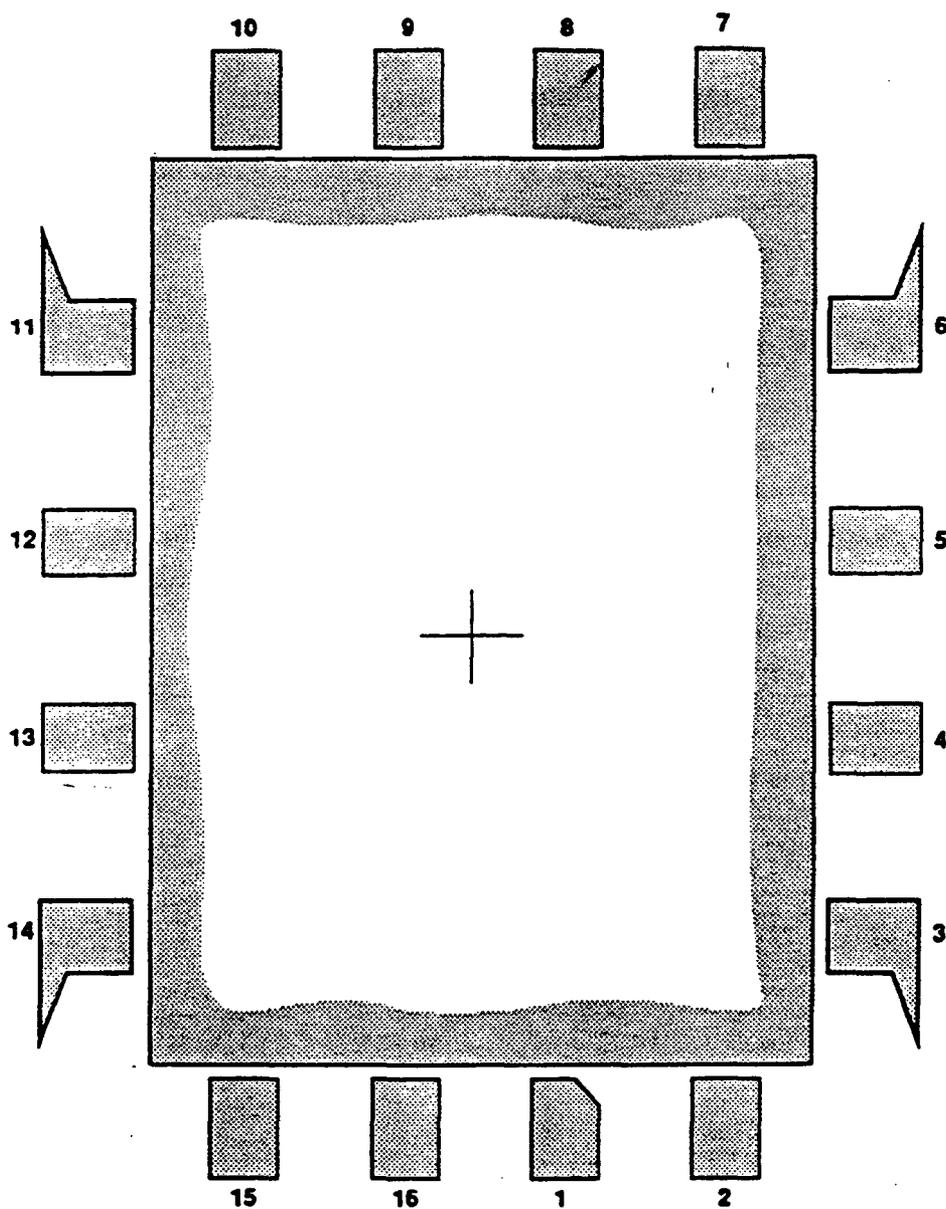


<p><b>TYPE</b> <b>DILC 20</b></p>	<p><b>CAVITY SIZE (mm)</b> <b>6.6 x 4.4</b></p>	<p><b>SCALE</b> <b>20</b></p>
---------------------------------------	---	-----------------------------------



<p><b>TYPE</b> <b>DILC 18</b></p>	<p><b>CAVITY SIZE (mm)</b> <b>5.6 x 4.3</b></p>	<p><b>SCALE</b> <b>20</b></p>
---------------------------------------	---	-----------------------------------

10. Package drawings:



<p><b>TYPE</b> <b>DILC 16</b></p>	<p><b>CAVITY SIZE (mm)</b> <b>6.1 x 4.3</b></p>	<p><b>SCALE</b> <b>20</b></p>
---------------------------------------	---	-----------------------------------

## **Apéndice II**

En este apéndice se incluyen los modelos de los distintos dispositivos que dispone esta tecnología. Así, en una serie de tablas se incluyen todos los parámetros necesarios para caracterizar a los dispositivos, ya sean transistores, condensadores o resistencias. Como se podrá observar, se dan tres tipos de parámetros, los máximos, los mínimos y los típicos, para de esta manera poder realizar los cálculos extremos si se desea.

### 3. Spice models for POLYUSE L transistors and diodes

#### QN1 spice model

	QN1 TYP NPN	QN1 MIN NPN	QN1 MAX NPN
IS	180.000000E-18	130.000000E-18	230.000000E-18
EG	1.205	1.205	1.205
BF	125	70	180
NF	1	1	1
VAF	35	48.6	27.3
IKF	.0115	.0145	.01
ISE	250.000000E-18	250.000000E-18	250.000000E-18
NE	1.3	1.3	1.3
BR	1.9	1.37	2.43
NR	1	1	1
VAR	5.8	7.7	4.8
IKR	6.000000E-03	7.600000E-03	5.200000E-03
RB	300	231	370
RBM	65	65	65
IRB	150.000000E-06	260.000000E-06	100.000000E-06
RE	5	5	5
RC	140	140	140
CJE	140.000000E-15	145.000000E-15	132.000000E-15
VJE	.85	.85	.85
MJE	.4	.4	.4
CJC	170.000000E-15	170.000000E-15	170.000000E-15
VJC	.6	.6	.6
MJC	.38	.38	.38
XCJC	.2	.2	.2
CJS	360.000000E-15	360.000000E-15	360.000000E-15
VJS	.67	.67	.67
MJS	.43	.43	.43
TF	47.000000E-12	51.400000E-12	42.600000E-12
WTF	1	1	1
ITF	5.000000E-03	5.000000E-03	5.000000E-03
TR	6.000000E-09	8.330000E-09	4.700000E-09
XTB	2.5	2.5	2.5
XTI	2.9	2.9	2.9
KF	10.000000E-15	10.000000E-15	10.000000E-15
AF	1.33	1.33	1.33

QPL spice model

	QPL TYP PNP	QPL MIN PNP	QPL MAX PNP
IS	58.000000E-18	58.000000E-18	58.000000E-18
EG	1.205	1.205	1.205
BF	50	35	60
NF	1	1	1
VAF	45	45	45
IKE	59.000000E-06	59.000000E-06	59.000000E-06
ISE	110.000000E-18	110.000000E-18	110.000000E-18
NE	1.4	1.4	1.4
BR	2	1.4	2.4
NR	1	1	1
VAR	22	22	22
IKR	30.000000E-06	30.000000E-06	30.000000E-06
ISC	57.000000E-18	57.000000E-18	57.000000E-18
NC	1.54	1.54	1.54
RB	365	350	350
RBM	25	25	25
IRB	1.200000E-06	1.200000E-06	1.200000E-06
RE	9	9	9
RC	8	8	8
CJE	11.000000E-15	11.000000E-15	11.000000E-15
VJE	.65	.65	.6
MJE	.36	.36	.36
CJC	190.000000E-15	190.000000E-15	190.000000E-15
VJC	.6	.6	.6
MJC	.36	.36	.36
FC	.95	.95	.95
TF	16.200000E-09	16.200000E-09	16.200000E-09
XTF	1	1	1
ITF	550.000000E-06	550.000000E-06	550.000000E-06
PTF	9	9	9
TR	90.000000E-09	90.000000E-09	90.000000E-09
XTB	2.2	2.2	2.2
XTI	2.6	2.6	2.6
AF	.1	.1	.1

QPL spice model

	QPL TYP PNP	QPL MIN PNP	QPL MAX PNP
IS	58.000000E-18	58.000000E-18	58.000000E-18
EG	1.205	1.205	1.205
BF	50	35	60
NF	1	1	1
VAF	45	45	45
IKE	59.000000E-06	59.000000E-06	59.000000E-06
ISE	110.000000E-18	110.000000E-18	110.000000E-18
NE	1.4	1.4	1.4
BR	2	1.4	2.4
NR	1	1	1
VAR	22	22	22
IKR	30.000000E-06	30.000000E-06	30.000000E-06
ISC	57.000000E-18	57.000000E-18	57.000000E-18
NC	1.54	1.54	1.54
RB	365	350	350
RBM	25	25	25
IRB	1.200000E-06	1.200000E-06	1.200000E-06
RE	9	9	9
RC	8	8	8
CJE	11.000000E-15	11.000000E-15	11.000000E-15
VJE	.65	.65	.6
MJE	.36	.36	.36
CJC	190.000000E-15	190.000000E-15	190.000000E-15
VJC	.6	.6	.6
MJC	.36	.36	.36
FC	.95	.95	.95
TF	16.200000E-09	16.200000E-09	16.200000E-09
XTF	1	1	1
ITF	550.000000E-06	550.000000E-06	550.000000E-06
PTF	9	9	9
TR	90.000000E-09	90.000000E-09	90.000000E-09
XTB	2.2	2.2	2.2
XTI	2.6	2.6	2.6
AF	.1	.1	.1

© Del documento, los autores. Digitalización realizada por ULPGC - Biblioteca Universitaria, 2006

Parameters terminology

	Name	Parameter	Unit
1	IS	Transport saturation current	A
2	BF	Ideal maximum forward beta	-
3	NF	Forward current emission coefficient	-
4	VAF	Forward early voltage	V
5	IKF	Corner for forward beta high current roll-off	A
6	ISE	B-E leakage saturation current	A
7	NE	B-E leakage emission coefficient	-
8	BR	Ideal maximum reverse beta	-
9	NR	Reverse current emission coefficient	-
10	VAR	Reverse early voltage	V
11	IKR	Corner for reverse beta high current roll-off	A
12	ISC	B-C leakage saturation current	A
13	NC	B-C leakage emission coefficient	-
14	RB	Zero bias base resistance	$\Omega$
15	IRB	Current where base resistance falls halfway to its min value	A
16	RBM	Minimum base resistance at high currents	$\Omega$
17	RE	Emitter resistance	$\Omega$
18	RC	Collector resistance in linear mode	$\Omega$
19	CJE	B-E zero-bias depletion capacitance	F
20	VJE	B-E built-in potential	V
21	MJE	B-E junction exponential factor	-
22	TF	Ideal forward transit time	s
23	XTF	Coefficient for bias dependence of TF	-
24	VTF	Voltage describing $V_{BC}$ dependence of TF	V
25	ITF	High current parameter for effect on TF	A
26	PTF	Excess phase at $\text{freq}=1.0/(TF^2\pi)$ Hz	deg
27	CJC	B-C zero bias depletion capacitance	F
28	VJC	B-C built-in potential	V
29	MJC	B-C junction exponential factor	-
30	XCJC	Fraction of B-C depletion capacitance connected to internal base mode	-
31	TR	Ideal reverse transit time	s
32	CJS	Zero-bias collector-substrate capacitance	F
33	VJS	Substrate junction built-in potential	V
34	MJS	Substrate junction exponential factor	-
35	XTB	Forward and reverse beta temperature exponent	-
36	EG	Energy gap for temperature effect on IS	eV
37	XTI	Temperature exponent for effect on IS	-
38	KF	Flicker noise exponent	-
39	AF	Flicker noise exponent	-
40	FC	Coefficient for forward-bias depletion capacitance formula	-

**QPS 025 spice model**

	QPS 025 TYP	QPS 025 MIN	QPS 025 MAX
IS	0.5E-14	0.5E-14	0.5E-14
BF	315	315	315
NF	1	1	1
VAF	36	36	36
IKF	3.25E-3	3.25E-3	3.25E-3
ISE	.55E-16	.55E-16	.55E-16
NE	1.2	1.2	1.2
BR	.1	.1	.1
NR	1	1	1
VAR	12	12	12
IKR	.325E-3	.325E-3	.325E-3
ISC	0	0	0
NC	2	2	2
RB	1000	1000	1000
IRB	75E-6	75E-6	75E-6
RBM	600	600	600
RE	1	1	1
RC	120	120	120
CJE	.25E-12	.25E-12	.25E-12
VJE	.6	.6	.6
MJE	.36	.36	.36
TF	3.55E-9	3.55E-9	3.55E-9
XTF	1	1	1
ITF	75E-3	75E-3	75E-3
PTF	0	0	0
CJC	.55E-12	.55E-12	.55E-12
VJC	.5	.5	.5
MJC	.3	.3	.3
XCJC	1	1	1
TR	.5E-6	.5E-6	.5E-6
XTB	2.2	2.2	2.2
EG	1.11	1.11	1.11
XTI	3	3	3
FC	0.5	0.5	0.5

**QNS 20 spice model**

	QNS 20 TYP	QNS 20 MIN	QNS 20 MAX
IS	36E-16	26E-16	46E-16
BF	125	70	180
NF	1	1	1
VAF	35	48.6	27.3
IKF	229E-3	289E-3	200E-3
ISE	5E-15	5E-15	5E-15
NE	1.3	1.3	1.3
BR	1.8	1.37	1.8
NR	1	1	1
VAR	5.8	7.7	4.8
IKR	50E-3	63E-3	43E-3
ISC	0	0	0
NC	2	2	2
RB	20	15	25
IRB	3E-3	5.2E-3	2E-3
RBM	4	4	4
RE	1.5	1.5	1.5
RC	10	10	10
CJE	3.5E-12	3.6E-12	3.3E-12
VJE	.85	.85	.85
MJE	.4	.4	.4
TF	47E-12	51.4E-12	47E-12
XTF	1	1	1
ITF	100E-3	100E-3	100E-3
CJC	6E-12	6E-12	6E-12
VJC	.6	.6	.6
MJC	.38	.38	.38
XCJC	.25	.25	.25
TR	6E-9	8E-9	4.7E-9
CJS	5.4E-12	5.4E-12	5.4E-12
VJS	.67	.67	.67
MJS	.43	.43	.43
XTB	2.5	2.5	2.5
EG	1.205	1.205	1.205
XTI	2.9	2.9	2.9

QN2 spice model

	QN2 TYP NPN	QN2 MIN NPN	QN2 MAX NPN
IS	420.000000E-18	300.000000E-18	548.000000E-18
EG	1.205	1.205	1.205
BF	125	70	180
NF	1	1	1
VAF	35	46.8	27.3
IKE	.023	.03	.0203
ISE	500.000000E-18	500.000000E-18	500.000000E-18
NE	1.3	1.3	1.3
BR	4.5	3.24	5.76
NR	1	1	1
VAR	5.8	7.7	4.8
IKR	7.000000E-03	8.900000E-03	6.000000E-03
RB	160	122	198
RBM	32	32	32
IRB	300.000000E-06	530.000000E-06	200.000000E-06
RE	7.5	7.5	7.5
RC	50	50	50
CJE	280.000000E-15	290.000000E-15	260.000000E-15
VJE	.85	.85	.85
MJE	.4	.4	.4
CJC	340.000000E-15	340.000000E-15	340.000000E-15
VJC	.6	.6	.6
MJC	.38	.38	.38
XCJC	.25	.25	.25
CJS	460.000000E-15	460.000000E-15	460.000000E-15
VJS	.67	.67	.67
MJS	.43	.43	.43
TF	47.000000E-12	51.400000E-12	42.600000E-12
XTF	1	1	1
ITF	.01	.01	.01
TR	3.000000E-09	4.200000E-09	2.340000E-09
XTB	2.5	2.5	2.5
XTI	2.9	2.9	2.9
KF	5.000000E-15	5.000000E-15	5.000000E-15
AF	1.33	1.33	1.33

QN4 spice model

	QN4 TYP NPN	QN4 MIN NPN	QN4 MAX NPN
IS	900.000000E-18	650.000000E-18	1.150000E-15
EG	1.205	1.205	1.205
BF	125	70	180
NF	1	1	1
VAF	35	48.6	27.3
IKE	.0573	.0725	.05
ISE	1.250000E-15	1.250000E-15	1.250000E-15
NE	1.3	1.3	1.3
BR	1.8	1.37	2.43
NR	1	1	1
VAR	5.8	7.7	4.8
IKR	.0125	.0158	.0108
RB	60	46.2	74
RBM	13	13	13
IRB	750.000000E-06	1.300000E-03	500.000000E-06
RE	3	3	3
RC	30	30	30
CJE	700.000000E-15	725.000000E-15	660.000000E-15
VJE	.85	.85	.85
MJE	.4	.4	.4
CJC	660.000000E-15	660.000000E-15	660.000000E-15
VJC	.6	.6	.6
MJC	.38	.38	.38
XCJC	.25	.25	.25
CJS	840.000000E-15	840.000000E-15	840.000000E-15
VJS	.67	.67	.67
MJS	.43	.43	.43
TF	47.000000E-12	51.400000E-12	42.600000E-12
XTF	1	1	1
ITF	.025	.025	.025
TR	6.000000E-09	8.000000E-09	4.690000E-09
XTB	2.5	2.5	2.5
XTI	2.9	2.9	2.9
KF			
AF			

## Apéndice III

En este apéndice vamos a mostrar el listado de la salida de una simulación y así poder observar todos los resultados que se obtienen, y los distintos cálculos para la realización de las simulaciones. En este caso, la simulación es un transitorio y el cálculo del consumo de la matriz de conmutación de alta frecuencia.

```
cma1> hspice listapot
```

```
Using: /home/soft/meta/H93A/h93a/sun4/hspice
```

```
ld.so: warning: /usr/lib/libc.so.1.6 has older revision than expected 8
```

```
***** HSPICE -- H93A.01 11:04:15 95/05/08 sun4
```

```
Copyright 1993 (C) by Meta-Software, Inc.
```

```
This computer program is protected by copyright law and
international treaties. Any dissemination or use of this
program, other than that permitted by Meta-Software, Inc.,
is unlawful and may result in prosecution under the law.
```

```
Input File: listapot
```

```
lic: USER: fmorera HOSTNAME: cma1 HOSTID: 401045d
```

```
lic: contacting server: yaiza
```

```
ld.so: warning: /usr/lib/libc.so.1.6 has older revision than expected 8
```

```
lic: Server permit path: /home/soft/meta/H93A/h93a/permit.hsp
```

```
lic: Site: las_palmas_u Created: 930327 Order#: 29012
```

```
lic: token number 352 allocated_SJF
```

```
Init: read install configuration file: /home/soft/meta/H93A/h93a/meta.cfg
```

```
*****
```

```
*****
```

```
.include librer
```

\* modelos pour polyuse.l fichier reduit pour tpenserg 27o290

\* model for small size npn n1

.model n1 npn

+is=1.8e-16 bf=130 nf=1 vaf=35 ikf=11.5e-3  
 +ise=0.25e-15 ne=1.6 br=1.8 nr=1 var=5 ikr=6e-3  
 +isc=0 nc=2 rb=300 irb=1.5e-4 rbm=65 re=7 rc=140  
 +cje=140e-15 vje=.85 mje=.4 tf=47e-12 xtf=1 itf=5e-3  
 +cjc=170e-15 vjc=.6 mjc=.38 xcjc=.2 tr=19e-9  
 +cjs=360e-15 vjs=.67 mjs=.43  
 +xtb=2.5 eg=1.205 xti=2.9  
 +af=1.33 kf=1e-14

\*

\* model for medium size npn n2

.model n2 npn

+is=3.6e-16 bf=130 nf=1 vaf=35 ikf=23e-3  
 +ise=0.5e-15 ne=1.6 br=2.9 nr=1 var=5 ikr=9.4e-3  
 +isc=0 nc=2 rb=160 irb=3e-4 rbm=32 re=4 rc=50  
 +cje=280e-15 vje=.85 mje=.4 tf=47e-12 xtf=1 itf=10e-3  
 +cjc=340e-15 vjc=.6 mjc=.38 xcjc=.25 tr=15e-9  
 +cjs=460e-15 vjs=.67 mjs=.43  
 +xtb=2.5 eg=1.205 xti=2.9  
 +af=1.33 kf=5e-15

\*

\* model for large size npn n4

.model n4 npn

+is=9e-16 bf=130 nf=1 vaf=35 ikf=57.3e-3  
 +ise=1.25e-15 ne=1.6 br=2.4 nr=1 var=5 ikr=12.5e-3  
 +isc=0 nc=2 rb=60 irb=7.5e-4 rbm=13 re=2.5 rc=30  
 +cje=700e-15 vje=.85 mje=.4 tf=47e-12 xtf=1 itf=25e-3  
 +cjc=660e-15 vjc=.6 mjc=.38 xcjc=.25 tr=18e-9

```

+cjs=840e-15 vjs=.67 mjs=.43
+xtb=2.5 eg=1.205 xti=2.9
*
* model for lateral pnp pl
.model pl pnp
+is=.58e-16 bf=71 nf=1 vaf=23 ikf=59e-6
+ise=1.1e-16 ne=1.52 br=10 nr=1 var=18
+ikr=30e-6 isc=57e-18 nc=1.77 rb=365 irb=1.2e-6
+rbb=25 re=9 rc=8 cje=11e-15 vje=.65
+mje=.25 tf=16e-9 xtf=1 itf=550e-6
+ptf=9 cjc=190e-15 vjc=.67 mjc=.43 xcjc=1
+tr=90e-9 xtb=2.2
+eg=1.205 xti=2.6 kf=0 af=0 fc=.95
*
*
* model for vertical substrate pnp ps
*
.model ps pnp
+is=2e-14 bf=300 nf=1 vaf=36 ikf=13e-3
+ise=2.22e-16 ne=1.2 br=1 nr=1 var=12
+ikr=1.3e-3 isc=0 nc=2 rb=270 irb=300e-6
+rbb=150 re=1e-2 rc=30 cje=1e-12 vje=.67
+mje=.43 tf=3.5e-9 xtf=1 itf=300e-3
+ptf=0 cjc=2.2e-12 vjc=.5 mjc=.3 xcjc=1
+tr=0.5e-6 xtb=2.2
+eg=1.11 xti=3 fc=.5
*
*
*model for large size npn n5
.model n5 npn
+is=9e-16 bf=130 nf=1 vaf=35 ikf=57.3e-3

```

+ise=1.25e-15 ne=1.6 br=2.4 nr=1 var=5 ikr=12.5e-3  
 +isc=0 nc=2 rb=60 irb=7.5e-4 rbm=13 re=2.5 rc=30  
 +cje=700e-15 vje=.85 mje=.4 tf=47e-12 xtf=1 itf=25e-3  
 +cjc=660e-15 vjc=.6 mjc=.38 xcjc=.25 tr=18e-9  
 +cjs=840e-15 vjs=.67 mjs=.43  
 +xtb=2.5 eg=1.205 xti=2.9

.options post	* net 24 = /r44.minus
* net 0 = gnd!	* net 25 = /q47.e
* net 1 = /i88.vs	* net 26 = /q40.b
* net 2 = /vcc	* net 27 = /r39.minus
* net 3 = /vee	* net 28 = /r51.plus
* net 4 = /vcmd0	* net 29 = /q59.b
* net 5 = /vcmd1	* net 30 = /q59.e
* net 6 = /vcontrol	* net 31 = /q62.e
* net 7 = /i0.vs	* net 32 = /q59.c
* net 8 = /i1.vs	* net 33 = /q62.c
* net 9 = /i2.vs	* net 34 = /q80.e
* net 10 = /i60.vs	* net 35 = /q21.e
* net 11 = /i76.vs	* net 36 = /q22.e
* net 12 = /q23.c	* net 37 = /q23.e
* net 13 = /q18.b	* net 38 = /q42.e
* net 14 = /q20.c	* net 39 = /q43.e
* net 15 = /q18.e	* net 40 = /q45.e
* net 16 = /q29.e	* net 41 = /q46.b
* net 17 = /q20.e	* net 42 = /q42.c
* net 18 = /q46.c	* net 43 = /q18.c
* net 19 = /r36.plus	* net 44 = /q43.c
* net 20 = /q48.c	* net 45 = /q75.e
* net 21 = /r36.minus	* net 46 = /q78.e
* net 22 = /q40.e	* net 47 = /q85.e
* net 23 = /r41.plus	* net 48 = /vin1

\* net 49 = /i64.vout  
 \* net 50 = /vs2  
 \* net 51 = /vs1  
 \* net 52 = /i12.vout  
 \* net 53 = /vin4  
 \* net 54 = /vin3  
 \* net 55 = /vin2  
 \* net 62 = /i34/q1.e  
 \* net 63 = /i34/q2.e  
 \* net 64 = /i34/q33.e  
 \* net 65 = /i34/q0.c  
 \* net 66 = /i34/q1.c  
 \* net 67 = /i34/r45.minus  
 \* net 68 = /i34/q31.b  
 \* net 69 = /i34/q0.e  
 \* net 70 = /i34/r23.minus  
 \* net 71 = /i34/q31.c  
 \* net 72 = /i34/r25.minus  
 \* net 73 = /i34/r27.minus  
 \* net 74 = /i34/q31.e  
 \* net 75 = /i34/q32.e  
 \* net 76 = /i34/r39.plus  
 \* net 77 = /i34/q2.c  
 \* net 78 = /i34/q3.c  
 \* net 79 = /i34/r41.plus  
 \* net 80 = /i34/q34.e  
 \* net 81 = /i34/q35.e  
 \* net 82 = /i34/r28.minus  
 \* net 83 = /i34/r12.minus  
 \* net 84 = /i34/r17.minus  
 \* net 85 = /i34/r15.minus  
 \* net 86 = /i34/r18.minus  
 \* net 87 = /i34/r16.minus  
 \* net 88 = /i34/r22.plus  
 \* net 89 = /i34/r21.plus  
 \* net 90 = /i34/r6.minus  
 \* resistor(0) = /i34/r30  
 r\$#0 82 88 200  
 \* resistor(1) = /i34/r8  
 r\$#1 90 2 4k  
 \* resistor(2) = /i34/r44  
 r\$#2 3 81 800  
 \* resistor(3) = /i34/r43  
 r\$#3 3 80 800  
 \* resistor(4) = /i34/r42  
 r\$#4 3 79 400  
 \* resistor(5) = /i34/r41  
 r\$#5 79 64 200  
 \* resistor(6) = /i34/r20  
 r\$#6 86 50 16k  
 \* resistor(7) = /i34/r19  
 r\$#7 84 51 16k  
 \* resistor(8) = /i34/r18  
 r\$#8 87 86 2k  
 \* resistor(9) = /i34/r17  
 r\$#9 85 84 2k  
 \* resistor(10) = /i34/r16  
 r\$#10 66 87 4k  
 \* resistor(11) = /i34/r15  
 r\$#11 65 85 4k  
 \* resistor(12) = /i34/r37  
 r\$#12 78 2 2k

\* resistor(13) = /i34/r36  
 r\$#13 77 2 2k  
 \* resistor(14) = /i34/r40  
 r\$#14 3 76 400  
 \* resistor(15) = /i34/r39  
 r\$#15 76 75 800  
 \* resistor(16) = /i34/r38  
 r\$#16 3 74 400  
 \* resistor(17) = /i34/r28  
 r\$#17 71 82 800  
 \* resistor(18) = /i34/r27  
 r\$#18 71 73 800  
 \* resistor(19) = /i34/r29  
 r\$#19 73 88 200  
 \* resistor(20) = /i34/r22  
 r\$#20 88 62 100  
 \* resistor(21) = /i34/r25  
 r\$#21 71 72 800  
 \* resistor(22) = /i34/r23  
 r\$#22 71 70 800  
 \* resistor(23) = /i34/r26  
 r\$#23 72 89 200  
 \* resistor(24) = /i34/r24  
 r\$#24 70 89 200  
 \* resistor(25) = /i34/r21  
 r\$#25 89 69 100  
 \* resistor(26) = /i34/r45  
 r\$#26 68 67 1k  
 \* resistor(27) = /i34/r46  
 r\$#27 67 2 8k  
 \* resistor(28) = /i34/r12

r\$#28 66 83 8k  
 \* resistor(29) = /i34/r14  
 r\$#29 83 2 4k  
 \* resistor(30) = /i34/r13  
 r\$#30 83 2 2k  
 \* resistor(31) = /i34/r11  
 r\$#31 66 2 4k  
 \* resistor(32) = /i34/r6  
 r\$#32 65 90 8k  
 \* resistor(33) = /i34/r7  
 r\$#33 90 2 2k  
 \* resistor(34) = /i34/r5  
 r\$#34 65 2 4k  
 \* qn2(35) = /i34/q5  
 q35 2 78 50 n2  
 \* qn2(36) = /i34/q4  
 q36 2 77 51 n2  
 \* qn1(37) = /i34/q35  
 q37 50 68 81 n1  
 \* qn1(38) = /i34/q34  
 q38 51 68 80 n1  
 \* qn1(39) = /i34/q33  
 q39 63 68 64 n1  
 \* qn1(40) = /i34/q32  
 q40 68 68 75 n1  
 \* qn1(41) = /i34/q31  
 q41 71 68 74 n1  
 \* qn1(42) = /i34/q3  
 q42 78 66 63 n1  
 \* qn1(43) = /i34/q2  
 q43 77 65 63 n1

\* qn1(44) = /i34/q1  
q44 66 20 62 n1  
\* qn1(45) = /i34/q0  
q45 65 18 69 n1  
\* qn1(46) = /q85  
q46 41 29 47 n1  
\* qn1(47) = /q80  
q47 2 33 34 n1  
\* qn1(48) = /q78  
q48 2 49 46 n1  
\* qn1(49) = /q75  
q49 29 29 45 n1  
\* qn1(50) = /q62  
q50 33 0 31 n1  
\* qn1(51) = /q59  
q51 32 29 30 n1  
\* qn1(52) = /q54  
q52 20 43 44 n1  
\* qn1(53) = /q49  
q53 18 43 42 n1  
\* qn1(54) = /q48  
q54 20 41 42 n1  
\* qn1(55) = /q47  
q55 39 26 25 n1  
\* qn1(56) = /q46  
q56 18 41 44 n1  
\* qn1(57) = /q45  
q57 38 26 40 n1  
\* qn1(58) = /q43  
q58 44 6 39 n1  
\* qn1(59) = /q42  
q59 42 0 38 n1  
\* qn1(60) = /q40  
q60 26 26 22 n1  
\* qn1(61) = /q29  
q61 2 14 16 n1  
\* qn1(62) = /q23  
q62 12 13 37 n1  
\* qn1(63) = /q22  
q63 13 13 36 n1  
\* qn1(64) = /q21  
q64 2 52 35 n1  
\* qn1(65) = /q20  
q65 14 0 17 n1  
\* qn1(66) = /q18  
q66 43 13 15 n1  
\* resistor(67) = /r83  
r\$#67 41 34 2k  
\* resistor(68) = /r81  
r\$#68 3 47 10k  
\* resistor(69) = /r79  
r\$#69 33 2 1.2k  
\* resistor(70) = /r77  
r\$#70 32 46 300  
\* resistor(71) = /r70  
r\$#71 32 31 300  
\* resistor(72) = /r67  
r\$#72 3 45 10k  
\* resistor(73) = /r63  
r\$#73 3 30 1k  
\* resistor(74) = /r61  
r\$#74 2 29 18k

\* resistor(75) = /r57  
r\$#75 23 3 100

\* resistor(76) = /r56  
r\$#76 3 28 100

\* resistor(77) = /r55  
r\$#77 27 26 100

\* resistor(78) = /r52  
r\$#78 24 25 400

\* resistor(79) = /r51  
r\$#79 28 40 400

\* resistor(80) = /r50  
r\$#80 39 38 300

\* resistor(81) = /r44  
r\$#81 3 24 100

\* resistor(82) = /r41  
r\$#82 23 22 400

\* resistor(83) = /r39  
r\$#83 21 27 200

\* resistor(84) = /r38  
r\$#84 20 2 1k1

\* resistor(85) = /r37  
r\$#85 0 19 1k

\* resistor(86) = /r36  
r\$#86 19 21 200

\* resistor(87) = /r33  
r\$#87 18 2 1k1

\* resistor(88) = /r28  
r\$#88 12 17 300

\* resistor(89) = /r27  
r\$#89 43 16 2k

\* resistor(90) = /r26  
r\$#90 3 37 1k

\* resistor(91) = /r25  
r\$#91 3 15 10k

\* resistor(92) = /r24  
r\$#92 14 2 1.2k

\* resistor(93) = /r19  
r\$#93 2 13 18k

\* resistor(94) = /r17  
r\$#94 3 36 10k

\* resistor(95) = /r15  
r\$#95 12 35 300

\* net 95 = /i64/q0.e

\* net 96 = /i64/q2.e

\* net 97 = /i64/q1.c

\* net 98 = /i64/q2.b

\* net 99 = /i64/q6.e

\* net 100 = /i64/q7.e

\* qn5(101) = /i64/q7  
q101 98 98 100 n5

\* qn5(102) = /i64/q6  
q102 49 98 99 n5

\* qn1(103) = /i64/q5  
q103 2 97 49 n1

\* resistor(104) = /i64/r8  
r\$#104 3 99 100

\* resistor(105) = /i64/r10  
r\$#105 98 2 13k

\* resistor(106) = /i64/r9  
r\$#106 3 100 2k

\* resistor(107) = /i64/r3  
r\$#107 2 97 7k

```

* resistor(108) = /i64/r4
r$#108 3 96 1k
* qn1(109) = /i64/q2
q109 95 98 96 n1
* qn1(110) = /i64/q1
q110 97 49 95 n1
* qn1(111) = /i64/q0
q111 2 11 95 n1
* net 105 = /i12/q0.e
* net 106 = /i12/q2.e
* net 107 = /i12/q1.c
* net 108 = /i12/q2.b
* net 109 = /i12/q6.e
* net 110 = /i12/q7.e
* qn5(112) = /i12/q7
q112 108 108 110 n5
* qn5(113) = /i12/q6
q113 52 108 109 n5
* qn1(114) = /i12/q5
q114 2 107 52 n1
* resistor(115) = /i12/r8
r$#115 3 109 100
* resistor(116) = /i12/r10
r$#116 108 2 13k
* resistor(117) = /i12/r9
r$#117 3 110 2k
* resistor(118) = /i12/r3
r$#118 2 107 7k
* resistor(119) = /i12/r4
r$#119 3 106 1k
* qn1(120) = /i12/q2
q120 105 108 106 n1
* qn1(121) = /i12/q1
q121 107 52 105 n1
* qn1(122) = /i12/q0
q122 2 9 105 n1
* net 116 = /i76/r2.plus
* net 117 = /i76/q4.b
* net 118 = /i76/r21.plus
* net 119 = /i76/q36.e
* net 120 = /i76/r14.minus
* net 121 = /i76/r0.minus
* net 122 = /i76/q40.e
* net 123 = /i76/q18.c
* net 124 = /i76/r26.minus
* net 125 = /i76/r39.minus
* net 126 = /i76/r15.plus
* net 127 = /i76/r38.plus
* net 128 = /i76/q4.e
* net 129 = /i76/q19.e
* net 130 = /i76/q19.c
* net 131 = /i76/q13.b
* net 132 = /i76/q36.c
* net 133 = /i76/q12.c
* net 134 = /i76/q27.e
* net 135 = /i76/q13.e
* net 136 = /i76/q18.e
* net 137 = /i76/q13.c
* net 138 = /i76/q25.e
* net 139 = /i76/q12.e
* net 140 = /i76/q30.c
* net 141 = /i76/q10.c

```

- \* net 142 = /i76/q18.b
- \* net 143 = /i76/r11.plus
- \* net 144 = /i76/r0.plus
- \* qn1(124) = /i76/q42  
q124 132 1 139 n1
- \* qn1(125) = /i76/q37  
q125 132 10 137 n1
- \* qn1(126) = /i76/q30  
q126 140 140 141 n1
- \* qn1(127) = /i76/q25  
q127 11 142 138 n1
- \* qn1(128) = /i76/q20  
q128 133 11 137 n1
- \* qn1(129) = /i76/q18  
q129 123 142 136 n1
- \* qn1(130) = /i76/q13  
q130 137 131 135 n1
- \* qn1(131) = /i76/q12  
q131 133 11 139 n1
- \* qn1(132) = /i76/q10  
q132 141 141 0 n1
- \* qn1(133) = /i76/q7  
q133 139 130 135 n1
- \* qn2(134) = /i76/q27  
q134 135 142 134 n2
- \* qn2(135) = /i76/q22  
q135 2 133 11 n2
- \* qn2(136) = /i76/q5  
q136 2 123 142 n2
- \* qpl(137) = /i76/q40  
q137 133 132 122 pl
- \* qpl(138) = /i76/q36  
q138 132 132 119 pl
- \* qpl(139) = /i76/q34  
q139 131 140 129 pl
- \* qpl(140) = /i76/q19  
q140 130 5 129 pl
- \* qpl(141) = /i76/q4  
q141 129 117 128 pl
- \* resistor(142) = /i76/r21  
r\$#142 118 117 4k
- \* resistor(143) = /i76/r8  
r\$#143 140 127 1k
- \* resistor(144) = /i76/r28  
r\$#144 121 3 100
- \* resistor(145) = /i76/r14  
r\$#145 144 120 400
- \* resistor(146) = /i76/r1  
r\$#146 3 138 200
- \* resistor(147) = /i76/r29  
r\$#147 3 126 200
- \* resistor(148) = /i76/r15  
r\$#148 126 138 100
- \* resistor(149) = /i76/r9  
r\$#149 0 125 200
- \* resistor(150) = /i76/r39  
r\$#150 124 125 1k
- \* resistor(151) = /i76/r26  
r\$#151 123 124 16k
- \* resistor(152) = /i76/r24  
r\$#152 136 3 800
- \* resistor(153) = /i76/r3

r\$#153 122 2 400  
 \* resistor(154) = /i76/r0  
 r\$#154 144 121 200  
 \* resistor(155) = /i76/r6  
 r\$#155 120 134 1k  
 \* resistor(156) = /i76/r32  
 r\$#156 119 2 400  
 \* resistor(157) = /i76/r38  
 r\$#157 127 118 2k  
 \* resistor(158) = /i76/r23  
 r\$#158 117 2 16k  
 \* resistor(159) = /i76/r31  
 r\$#159 3 143 16k  
 \* resistor(160) = /i76/r11  
 r\$#160 143 131 2k  
 \* resistor(161) = /i76/r33  
 r\$#161 130 143 2k  
 \* resistor(162) = /i76/r2  
 r\$#162 116 2 2k  
 \* resistor(163) = /i76/r35  
 r\$#163 128 116 16k  
 \* net 151 = /i2/r2.plus  
 \* net 152 = /i2/q4.b  
 \* net 153 = /i2/r21.plus  
 \* net 154 = /i2/q36.e  
 \* net 155 = /i2/r14.minus  
 \* net 156 = /i2/r0.minus  
 \* net 157 = /i2/q40.e  
 \* net 158 = /i2/q18.c  
 \* net 159 = /i2/r26.minus  
 \* net 160 = /i2/r39.minus  
 \* net 161 = /i2/r15.plus  
 \* net 162 = /i2/r38.plus  
 \* net 163 = /i2/q4.e  
 \* net 164 = /i2/q19.e  
 \* net 165 = /i2/q19.c  
 \* net 166 = /i2/q13.b  
 \* net 167 = /i2/q36.c  
 \* net 168 = /i2/q12.c  
 \* net 169 = /i2/q27.e  
 \* net 170 = /i2/q13.e  
 \* net 171 = /i2/q18.e  
 \* net 172 = /i2/q13.c  
 \* net 173 = /i2/q25.e  
 \* net 174 = /i2/q12.e  
 \* net 175 = /i2/q30.c  
 \* net 176 = /i2/q10.c  
 \* net 177 = /i2/q18.b  
 \* net 178 = /i2/r11.plus  
 \* net 179 = /i2/r0.plus  
 \* qn1(165) = /i2/q42  
 q165 167 7 174 n1  
 \* qn1(166) = /i2/q37  
 q166 167 8 172 n1  
 \* qn1(167) = /i2/q30  
 q167 175 175 176 n1  
 \* qn1(168) = /i2/q25  
 q168 9 177 173 n1  
 \* qn1(169) = /i2/q20  
 q169 168 9 172 n1  
 \* qn1(170) = /i2/q18  
 q170 158 177 171 n1

\* qn1(171) = /i2/q13  
q171 172 166 170 n1  
\* qn1(172) = /i2/q12  
q172 168 9 174 n1  
\* qn1(173) = /i2/q10  
q173 176 176 0 n1  
\* qn1(174) = /i2/q7  
q174 174 165 170 n1  
\* qn2(175) = /i2/q27  
q175 170 177 169 n2  
\* qn2(176) = /i2/q22  
q176 2 168 9 n2  
\* qn2(177) = /i2/q5  
q177 2 158 177 n2  
\* qpl(178) = /i2/q40  
q178 168 167 157 pl  
\* qpl(179) = /i2/q36  
q179 167 167 154 pl  
\* qpl(180) = /i2/q34  
q180 166 175 164 pl  
\* qpl(181) = /i2/q19  
q181 165 5 164 pl  
\* qpl(182) = /i2/q4  
q182 164 152 163 pl  
\* resistor(183) = /i2/r21  
r\$#183 153 152 4k  
\* resistor(184) = /i2/r8  
r\$#184 175 162 1k  
\* resistor(185) = /i2/r28  
r\$#185 156 3 100  
\* resistor(186) = /i2/r14  
r\$#186 179 155 400  
\* resistor(187) = /i2/r1  
r\$#187 3 173 200  
\* resistor(188) = /i2/r29  
r\$#188 3 161 200  
\* resistor(189) = /i2/r15  
r\$#189 161 173 100  
\* resistor(190) = /i2/r9  
r\$#190 0 160 200  
\* resistor(191) = /i2/r39  
r\$#191 159 160 1k  
\* resistor(192) = /i2/r26  
r\$#192 158 159 16k  
\* resistor(193) = /i2/r24  
r\$#193 171 3 800  
\* resistor(194) = /i2/r3  
r\$#194 157 2 400  
\* resistor(195) = /i2/r0  
r\$#195 179 156 200  
\* resistor(196) = /i2/r6  
r\$#196 155 169 1k  
\* resistor(197) = /i2/r32  
r\$#197 154 2 400  
\* resistor(198) = /i2/r38  
r\$#198 162 153 2k  
\* resistor(199) = /i2/r23  
r\$#199 152 2 16k  
\* resistor(200) = /i2/r31  
r\$#200 3 178 16k  
\* resistor(201) = /i2/r11  
r\$#201 178 166 2k

\* resistor(202) = /i2/r33  
 r\$#202 165 178 2k  
 \* resistor(203) = /i2/r2  
 r\$#203 151 2 2k  
 \* resistor(204) = /i2/r35  
 r\$#204 163 151 16k  
 \* net 186 = /i88/q0.b  
 \* net 187 = /i88/q1.b  
 \* net 188 = /i88/q12.e  
 \* net 189 = /i88/q12.c  
 \* net 190 = /i88/q19.c  
 \* net 191 = /i88/q2.e  
 \* net 192 = /i88/q11.c  
 \* net 193 = /i88/q13.e  
 \* net 194 = /i88/q40.e  
 \* net 195 = /i88/q43.e  
 \* net 196 = /i88/q11.e  
 \* net 197 = /i88/q4.e  
 \* net 198 = /i88/q40.c  
 \* net 199 = /i88/r16.minus  
 \* net 200 = /i88/q5.b  
 \* net 201 = /i88/r22.minus  
 \* net 202 = /i88/r26.plus  
 \* net 203 = /i88/r15.minus  
 \* net 204 = /i88/q5.e  
 \* net 205 = /i88/r24.plus  
 \* net 206 = /i88/r17.plus  
 \* net 207 = /i88/q1.e  
 \* net 208 = /i88/r36.plus  
 \* net 209 = /i88/r21.minus  
 \* net 210 = /i88/q14.c

\* net 211 = /i88/q14.e  
 \* net 212 = /i88/q11.b  
 \* net 213 = /i88/r25.minus  
 \* net 214 = /i88/r16.plus  
 \* resistor(205) = /i88/r38  
 r\$#205 3 195 100  
 \* resistor(206) = /i88/r21  
 r\$#206 3 209 100  
 \* resistor(207) = /i88/r46  
 r\$#207 209 195 100  
 \* resistor(208) = /i88/r42  
 r\$#208 210 208 1k  
 \* resistor(209) = /i88/r41  
 r\$#209 207 2 400  
 \* resistor(210) = /i88/r39  
 r\$#210 213 190 2k  
 \* resistor(211) = /i88/r36  
 r\$#211 208 206 2k  
 \* resistor(212) = /i88/r35  
 r\$#212 214 205 200  
 \* resistor(213) = /i88/r32  
 r\$#213 204 202 16k  
 \* resistor(214) = /i88/r30  
 r\$#214 3 213 16k  
 \* resistor(215) = /i88/r29  
 r\$#215 194 3 800  
 \* resistor(216) = /i88/r28  
 r\$#216 203 201 1k  
 \* resistor(217) = /i88/r26  
 r\$#217 202 2 2k  
 \* resistor(218) = /i88/r25

r\$#218 189 213 2k  
 \* resistor(219) = /i88/r24  
 r\$#219 205 3 100  
 \* resistor(220) = /i88/r22  
 r\$#220 0 201 200  
 \* resistor(221) = /i88/r20  
 r\$#221 200 2 16k  
 \* resistor(222) = /i88/r17  
 r\$#222 206 200 4k  
 \* resistor(223) = /i88/r16  
 r\$#223 214 199 400  
 \* resistor(224) = /i88/r15  
 r\$#224 198 203 16k  
 \* resistor(225) = /i88/r8  
 r\$#225 197 2 400  
 \* resistor(226) = /i88/r7  
 r\$#226 199 196 1k  
 \* qn1(228) = /i88/q44  
 q228 191 190 192 n1  
 \* qn1(229) = /i88/q43  
 q229 1 212 195 n1  
 \* qn1(230) = /i88/q40  
 q230 198 212 194 n1  
 \* qn1(231) = /i88/q37  
 q231 211 211 0 n1  
 \* qn1(232) = /i88/q27  
 q232 187 0 193 n1  
 \* qn1(233) = /i88/q23  
 q233 187 0 191 n1  
 \* qn1(234) = /i88/q18  
 q234 193 189 192 n1  
 \* qn1(235) = /i88/q14  
 q235 210 210 211 n1  
 \* qn1(236) = /i88/q13  
 q236 186 1 193 n1  
 \* qn1(237) = /i88/q2  
 q237 186 1 191 n1  
 \* qpl(238) = /i88/q19  
 q238 190 210 188 pl  
 \* qpl(239) = /i88/q12  
 q239 189 4 188 pl  
 \* qpl(240) = /i88/q5  
 q240 188 200 204 pl  
 \* qpl(241) = /i88/q4  
 q241 187 187 197 pl  
 \* qpl(242) = /i88/q1  
 q242 186 187 207 pl  
 \* qn2(243) = /i88/q11  
 q243 192 212 196 n2  
 \* qn2(244) = /i88/q6  
 q244 2 198 212 n2  
 \* qn2(245) = /i88/q0  
 q245 2 186 1 n2  
 \* net 221 = /i60/q0.b  
 \* net 222 = /i60/q1.b  
 \* net 223 = /i60/q12.e  
 \* net 224 = /i60/q12.c  
 \* net 225 = /i60/q19.c  
 \* net 226 = /i60/q2.e  
 \* net 227 = /i60/q11.c  
 \* net 228 = /i60/q13.e  
 \* net 229 = /i60/q40.e

* net 230 = /i60/q43.e	r\$#251 248 225 2k
* net 231 = /i60/q11.e	* resistor(252) = /i60/r36
* net 232 = /i60/q4.e	r\$#252 243 241 2k
* net 233 = /i60/q40.c	* resistor(253) = /i60/r35
* net 234 = /i60/r16.minus	r\$#253 249 240 200
* net 235 = /i60/q5.b	* resistor(254) = /i60/r32
* net 236 = /i60/r22.minus	r\$#254 239 237 16k
* net 237 = /i60/r26.plus	* resistor(255) = /i60/r30
* net 238 = /i60/r15.minus	r\$#255 3 248 16k
* net 239 = /i60/q5.e	* resistor(256) = /i60/r29
* net 240 = /i60/r24.plus	r\$#256 229 3 800
* net 241 = /i60/r17.plus	* resistor(257) = /i60/r28
* net 242 = /i60/q1.e	r\$#257 238 236 1k
* net 243 = /i60/r36.plus	* resistor(258) = /i60/r26
* net 244 = /i60/r21.minus	r\$#258 237 2 2k
* net 245 = /i60/q14.c	* resistor(259) = /i60/r25
* net 246 = /i60/q14.e	r\$#259 224 248 2k
* net 247 = /i60/q11.b	* resistor(260) = /i60/r24
* net 248 = /i60/r25.minus	r\$#260 240 3 100
* net 249 = /i60/r16.plus	* resistor(261) = /i60/r22
* resistor(246) = /i60/r38	r\$#261 0 236 200
r\$#246 3 230 100	* resistor(262) = /i60/r20
* resistor(247) = /i60/r21	r\$#262 235 2 16k
r\$#247 3 244 100	* resistor(263) = /i60/r17
* resistor(248) = /i60/r46	r\$#263 241 235 4k
r\$#248 244 230 100	* resistor(264) = /i60/r16
* resistor(249) = /i60/r42	r\$#264 249 234 400
r\$#249 245 243 1k	* resistor(265) = /i60/r15
* resistor(250) = /i60/r41	r\$#265 233 238 16k
r\$#250 242 2 400	* resistor(266) = /i60/r8
* resistor(251) = /i60/r39	r\$#266 232 2 400

\* resistor(267) = /i60/r7  
 r\$#267 234 231 1k  
 \* qn1(269) = /i60/q44  
 q269 226 225 227 n1  
 \* qn1(270) = /i60/q43  
 q270 10 247 230 n1  
 \* qn1(271) = /i60/q40  
 q271 233 247 229 n1  
 \* qn1(272) = /i60/q37  
 q272 246 246 0 n1  
 \* qn1(273) = /i60/q27  
 q273 222 0 228 n1  
 \* qn1(274) = /i60/q23  
 q274 222 0 226 n1  
 \* qn1(275) = /i60/q18  
 q275 228 224 227 n1  
 \* qn1(276) = /i60/q14  
 q276 245 245 246 n1  
 \* qn1(277) = /i60/q13  
 q277 221 10 228 n1  
 \* qn1(278) = /i60/q2  
 q278 221 10 226 n1  
 \* qpl(279) = /i60/q19  
 q279 225 245 223 pl  
 \* qpl(280) = /i60/q12  
 q280 224 4 223 pl  
 \* qpl(281) = /i60/q5  
 q281 223 235 239 pl  
 \* qpl(282) = /i60/q4  
 q282 222 222 232 pl  
 \* qpl(283) = /i60/q1  
 q283 221 222 242 pl  
 \* qn2(284) = /i60/q11  
 q284 227 247 231 n2  
 \* qn2(285) = /i60/q6  
 q285 2 233 247 n2  
 \* qn2(286) = /i60/q0  
 q286 2 221 10 n2  
 \* net 256 = /i1/q0.b  
 \* net 257 = /i1/q1.b  
 \* net 258 = /i1/q12.e  
 \* net 259 = /i1/q12.c  
 \* net 260 = /i1/q19.c  
 \* net 261 = /i1/q2.e  
 \* net 262 = /i1/q11.c  
 \* net 263 = /i1/q13.e  
 \* net 264 = /i1/q40.e  
 \* net 265 = /i1/q43.e  
 \* net 266 = /i1/q11.e  
 \* net 267 = /i1/q4.e  
 \* net 268 = /i1/q40.c  
 \* net 269 = /i1/r16.minus  
 \* net 270 = /i1/q5.b  
 \* net 271 = /i1/r22.minus  
 \* net 272 = /i1/r26.plus  
 \* net 273 = /i1/r15.minus  
 \* net 274 = /i1/q5.e  
 \* net 275 = /i1/r24.plus  
 \* net 276 = /i1/r17.plus  
 \* net 277 = /i1/q1.e  
 \* net 278 = /i1/r36.plus  
 \* net 279 = /i1/r21.minus

\* net 280 = /i1/q14.c  
 \* net 281 = /i1/q14.e  
 \* net 282 = /i1/q11.b  
 \* net 283 = /i1/r25.minus  
 \* net 284 = /i1/r16.plus  
 \* resistor(287) = /i1/r38  
 r\$#287 3 265 100  
 \* resistor(288) = /i1/r21  
 r\$#288 3 279 100  
 \* resistor(289) = /i1/r46  
 r\$#289 279 265 100  
 \* resistor(290) = /i1/r42  
 r\$#290 280 278 1k  
 \* resistor(291) = /i1/r41  
 r\$#291 277 2 400  
 \* resistor(292) = /i1/r39  
 r\$#292 283 260 2k  
 \* resistor(293) = /i1/r36  
 r\$#293 278 276 2k  
 \* resistor(294) = /i1/r35  
 r\$#294 284 275 200  
 \* resistor(295) = /i1/r32  
 r\$#295 274 272 16k  
 \* resistor(296) = /i1/r30  
 r\$#296 3 283 16k  
 \* resistor(297) = /i1/r29  
 r\$#297 264 3 800  
 \* resistor(298) = /i1/r28  
 r\$#298 273 271 1k  
 \* resistor(299) = /i1/r26  
 r\$#299 272 2 2k  
 \* resistor(300) = /i1/r25  
 r\$#300 259 283 2k  
 \* resistor(301) = /i1/r24  
 r\$#301 275 3 100  
 \* resistor(302) = /i1/r22  
 r\$#302 0 271 200  
 \* resistor(303) = /i1/r20  
 r\$#303 270 2 16k  
 \* resistor(304) = /i1/r17  
 r\$#304 276 270 4k  
 \* resistor(305) = /i1/r16  
 r\$#305 284 269 400  
 \* resistor(306) = /i1/r15  
 r\$#306 268 273 16k  
 \* resistor(307) = /i1/r8  
 r\$#307 267 2 400  
 \* resistor(308) = /i1/r7  
 r\$#308 269 266 1k  
 \* qn1(310) = /i1/q44  
 q310 261 260 262 n1  
 \* qn1(311) = /i1/q43  
 q311 8 282 265 n1  
 \* qn1(312) = /i1/q40  
 q312 268 282 264 n1  
 \* qn1(313) = /i1/q37  
 q313 281 281 0 n1  
 \* qn1(314) = /i1/q27  
 q314 257 54 263 n1  
 \* qn1(315) = /i1/q23  
 q315 257 53 261 n1  
 \* qn1(316) = /i1/q18

q316 263 259 262 n1	* net 299 = /i0/q40.e
* qn1(317) = /i1/q14	* net 300 = /i0/q43.e
q317 280 280 281 n1	* net 301 = /i0/q11.e
* qn1(318) = /i1/q13	* net 302 = /i0/q4.e
q318 256 8 263 n1	* net 303 = /i0/q40.c
* qn1(319) = /i1/q2	* net 304 = /i0/r16.minus
q319 256 8 261 n1	* net 305 = /i0/q5.b
* qpl(320) = /i1/q19	* net 306 = /i0/r22.minus
q320 260 280 258 pl	* net 307 = /i0/r26.plus
* qpl(321) = /i1/q12	* net 308 = /i0/r15.minus
q321 259 4 258 pl	* net 309 = /i0/q5.e
* qpl(322) = /i1/q5	* net 310 = /i0/r24.plus
q322 258 270 274 pl	* net 311 = /i0/r17.plus
* qpl(323) = /i1/q4	* net 312 = /i0/q1.e
q323 257 257 267 pl	* net 313 = /i0/r36.plus
* qpl(324) = /i1/q1	* net 314 = /i0/r21.minus
q324 256 257 277 pl	* net 315 = /i0/q14.c
* qn2(325) = /i1/q11	* net 316 = /i0/q14.e
q325 262 282 266 n2	* net 317 = /i0/q11.b
* qn2(326) = /i1/q6	* net 318 = /i0/r25.minus
q326 2 268 282 n2	* net 319 = /i0/r16.plus
* qn2(327) = /i1/q0	* resistor(328) = /i0/r38
q327 2 256 8 n2	r\$#328 3 300 100
* net 291 = /i0/q0.b	* resistor(329) = /i0/r21
* net 292 = /i0/q1.b	r\$#329 3 314 100
* net 293 = /i0/q12.e	* resistor(330) = /i0/r46
* net 294 = /i0/q12.c	r\$#330 314 300 100
* net 295 = /i0/q19.c	* resistor(331) = /i0/r42
* net 296 = /i0/q2.e	r\$#331 315 313 1k
* net 297 = /i0/q11.c	* resistor(332) = /i0/r41
* net 298 = /i0/q13.e	r\$#332 312 2 400

\* resistor(333) = /i0/r39  
r\$#333 318 295 2k  
\* resistor(334) = /i0/r36  
r\$#334 313 311 2k  
\* resistor(335) = /i0/r35  
r\$#335 319 310 200  
\* resistor(336) = /i0/r32  
r\$#336 309 307 16k  
\* resistor(337) = /i0/r30  
r\$#337 3 318 16k  
\* resistor(338) = /i0/r29  
r\$#338 299 3 800  
\* resistor(339) = /i0/r28  
r\$#339 308 306 1k  
\* resistor(340) = /i0/r26  
r\$#340 307 2 2k  
\* resistor(341) = /i0/r25  
r\$#341 294 318 2k  
\* resistor(342) = /i0/r24  
r\$#342 310 3 100  
\* resistor(343) = /i0/r22  
r\$#343 0 306 200  
\* resistor(344) = /i0/r20  
r\$#344 305 2 16k  
\* resistor(345) = /i0/r17  
r\$#345 311 305 4k  
\* resistor(346) = /i0/r16  
r\$#346 319 304 400  
\* resistor(347) = /i0/r15  
r\$#347 303 308 16k  
\* resistor(348) = /i0/r8  
r\$#348 302 2 400  
\* resistor(349) = /i0/r7  
r\$#349 304 301 1k  
\* qn1(351) = /i0/q44  
q351 296 295 297 n1  
\* qn1(352) = /i0/q43  
q352 7 317 300 n1  
\* qn1(353) = /i0/q40  
q353 303 317 299 n1  
\* qn1(354) = /i0/q37  
q354 316 316 0 n1  
\* qn1(355) = /i0/q27  
q355 292 48 298 n1  
\* qn1(356) = /i0/q23  
q356 292 55 296 n1  
\* qn1(357) = /i0/q18  
q357 298 294 297 n1  
\* qn1(358) = /i0/q14  
q358 315 315 316 n1  
\* qn1(359) = /i0/q13  
q359 291 7 298 n1  
\* qn1(360) = /i0/q2  
q360 291 7 296 n1  
\* qpl(361) = /i0/q19  
q361 295 315 293 pl  
\* qpl(362) = /i0/q12  
q362 294 4 293 pl  
\* qpl(363) = /i0/q5  
q363 293 305 309 pl  
\* qpl(364) = /i0/q4  
q364 292 292 302 pl

```

* qpl(365) = /i0/q1
q365 291 292 312 pl
* qn2(366) = /i0/q11
q366 297 317 301 n2
* qn2(367) = /i0/q6
q367 2 303 317 n2
* qn2(368) = /i0/q0
q368 2 291 7 n2
vcc 2 0 7.5
vee 0 3 7.5
vcmd0 4 0 5
vcmd1 5 0 0
vin1 48 0 ac .02 sin(0 .02 10e6)
vin2 55 0 ac .04 sin(0 .04 5e6)
vin3 54 0 ac .01 sin(0 .01 2e6)
vin4 53 0 ac .03 sin(0 .03 12e6)
vcontrol 6 0 pwl(0 0, 1.5u .3, 3u 0)
.tran .001u 3u
.meas tran idd rms i(vcc) from=0 to 3u
.end

```

```

***** HSPICE -- H93A.01      11:04:15 95/05/08  sun4

```

```

*****

```

```

***** bjt model parameters      tnom= 25.000 temp= 25.000

```

```

*****

```

```

*****

```

```

*** bjt model parameters  model name: 0:n1      model type:npn  ***

```

\*\*\*\*\*

names	values	units	names	values	units	names	values	units
-----	-----	-----	-----	-----	-----	-----	-----	-----

1\*\*\* basic dc parameters \*\*\*

level=	1.00		bf=	130.00		br=	1.80	
brs=	0.		bulk=	gnd		is=	1.8e-16	amps
iss=	0.	amps	nf=	1.00		nr=	1.00	
ns=	1.00		ibe=	0.	amps	ibc=	0.	amps
subs=	1.00		expli=	1.0e+15	amps			

2\*\*\* low current beta degradation effect parameters \*\*\*

isc=	0.	amps	ise=	2.5e-16	amps	nc=	2.00	
ne=	1.60							

3\*\*\* base width modulation parameters \*\*\*

vaf=	35.00	volts	var=	5.00	volts
------	-------	-------	------	------	-------

4\*\*\* high current beta degradation effect parameters \*\*\*

ikf=	11.50m	amps	nkf=	500.00m		ikr=	6.00m	amps
------	--------	------	------	---------	--	------	-------	------

5\*\*\* parasitic resistor parameters \*\*\*

irb=	150.00u	amps	rb=	300.00	ohms	rbm=	65.00	ohms
re=	7.00	ohms	rc=	140.00	ohms	vo=	0.	volts
gamma=	0.							

6\*\*\* junction capacitor parameters \*\*\*

cbcp=	0.	farad	cbep=	0.	farad	ccsp=	0.	farad
cjc=	170.00f	farad	cje=	140.00f	farad	cjs=	360.00f	farad
fc=	500.00m		mjc=	380.00m		mje=	400.00m	

mjs= 430.00m      vjc= 600.00m volts      vje= 850.00m volts  
 vjs= 670.00m volts      xcjc= 200.00m      qco= 0. coul

7\*\*\* transit time parameters \*\*\*

itf= 5.00m amps      ptf= 0. deg k      tf= 47.00p secs  
 tr= 19.00n secs      vtf= 0.      xtf= 1.00

8\*\*\* temperature compensation parameters \*\*\*

tlev= 0.      tlevc= 0.      trel= 0. /deg  
 tre2= 0. /deg2      trb1= 0. /deg      trc1= 0. /deg  
 trb2= 0. /deg2      trm1= 0. /deg      xtb= 2.50  
 trm2= 0. /deg2      xti= 2.90      cte= 0. /deg  
 ctc= 0. /deg      cts= 0. /deg      trc2= 0. /deg2  
 tref= 25.00 deg c      bex= 2.42      bexv= 1.90

9\*\*\* noise parameters \*\*\*

kf= 10.00f      af= 1.33

\*\*\*\*\*

\*\*\* bjt model parameters model name: 0:n2 model type:npn \*\*\*

\*\*\*\*\*

names	values	units	names	values	units	names	values	units
-------	--------	-------	-------	--------	-------	-------	--------	-------

-----

1\*\*\* basic dc parameters \*\*\*

level= 1.00      bf= 130.00      br= 2.90  
 brs= 0.      bulk= gnd      is= 3.6e-16 amps  
 iss= 0. amps      nf= 1.00      nr= 1.00  
 ns= 1.00      ibe= 0. amps      ibc= 0. amps  
 subs= 1.00      expli= 1.0e+15 amps

2\*\*\* low current beta degradation effect parameters \*\*\*

isc= 0. amps      ise= 5.0e-16 amps      nc= 2.00  
ne= 1.60

3\*\*\* base width modulation parameters \*\*\*

vaf= 35.00 volts      var= 5.00 volts

4\*\*\* high current beta degradation effect parameters \*\*\*

ikf= 23.00m amps      nkf= 500.00m      ikr= 9.40m amps

5\*\*\* parasitic resistor parameters \*\*\*

irb= 300.00u amps      rb= 160.00 ohms      rbm= 32.00 ohms  
re= 4.00 ohms      rc= 50.00 ohms      vo= 0. volts  
gamma= 0.

6\*\*\* junction capacitor parameters \*\*\*

cbcp= 0. farad      cbep= 0. farad      ccsp= 0. farad  
cjc= 340.00f farad      cje= 280.00f farad      cjs= 460.00f farad  
fc= 500.00m      mjc= 380.00m      mje= 400.00m  
mjs= 430.00m      vjc= 600.00m volts      vje= 850.00m volts  
vjs= 670.00m volts      xcjc= 250.00m      qco= 0. coul

7\*\*\* transit time parameters \*\*\*

itf= 10.00m amps      ptf= 0. deg k      tf= 47.00p secs  
tr= 15.00n secs      vtf= 0.      xtf= 1.00

8\*\*\* temperature compensation parameters \*\*\*

tlev= 0.      tlevc= 0.      tre1= 0. /deg  
tre2= 0. /deg2      trb1= 0. /deg      trc1= 0. /deg  
trb2= 0. /deg2      trm1= 0. /deg      xtb= 2.50  
trm2= 0. /deg2      xti= 2.90      cte= 0. /deg

ctc= 0. /deg    cts= 0. /deg    trc2= 0. /deg2  
 tref= 25.00 deg c    bex= 2.42    bexv= 1.90

9\*\*\* noise parameters \*\*\*

kf= 5.00f    af= 1.33

\*\*\*\*\*

\*\*\* bjt model parameters    model name: 0:n4    model type:npn \*\*\*

\*\*\*\*\*

names	values	units	names	values	units	names	values	units
----	-----	-----	----	-----	-----	----	-----	-----

1\*\*\* basic dc parameters \*\*\*

level= 1.00    bf= 130.00    br= 2.40  
 brs= 0.    bulk= gnd    is= 9.0e-16 amps  
 iss= 0. amps    nf= 1.00    nr= 1.00  
 ns= 1.00    ibe= 0. amps    ibc= 0. amps  
 subs= 1.00    expli= 1.0e+15 amps

2\*\*\* low current beta degradation effect parameters \*\*\*

isc= 0. amps    ise= 1.25f amps    nc= 2.00  
 ne= 1.60

3\*\*\* base width modulation parameters \*\*\*

vaf= 35.00 volts    var= 5.00 volts

4\*\*\* high current beta degradation effect parameters \*\*\*

ikf= 57.30m amps    nkf= 500.00m    ikr= 12.50m amps

5\*\*\* parasitic resistor parameters \*\*\*

irb= 750.00u amps      rb= 60.00 ohms      rbm= 13.00 ohms  
 re= 2.50 ohms      rc= 30.00 ohms      vo= 0. volts  
 gamma= 0.

6\*\*\* junction capacitor parameters \*\*\*

cbcp= 0. farad      cbep= 0. farad      ccsp= 0. farad  
 cjc= 660.00f farad      cje= 700.00f farad      cjs= 840.00f farad  
 fc= 500.00m      mjc= 380.00m      mje= 400.00m  
 mjs= 430.00m      vjc= 600.00m volts      vje= 850.00m volts  
 vjs= 670.00m volts      xcjc= 250.00m      qco= 0. coul

7\*\*\* transit time parameters \*\*\*

itf= 25.00m amps      ptf= 0. deg k      tf= 47.00p secs  
 tr= 18.00n secs      vtf= 0.      xtf= 1.00

8\*\*\* temperature compensation parameters \*\*\*

tlev= 0.      tlevc= 0.      trel= 0. /deg  
 tre2= 0. /deg2      trb1= 0. /deg      trc1= 0. /deg  
 trb2= 0. /deg2      trm1= 0. /deg      xtb= 2.50  
 trm2= 0. /deg2      xti= 2.90      cte= 0. /deg  
 ctc= 0. /deg      cts= 0. /deg      trc2= 0. /deg2  
 tref= 25.00 deg c      bex= 2.42      bexv= 1.90

9\*\*\* noise parameters \*\*\*

kf= 0.      af= 1.00

\*\*\*\*\*

\*\*\* bjt model parameters model name: 0:pl model type:npn \*\*\*

\*\*\*\*\*

names values units names values units names values units

1\*\*\* basic dc parameters \*\*\*

level= 1.00            bf= 71.00            br= 10.00  
 brs= 0.            bulk= gnd            is= 5.8e-17 amps  
 iss= 0. amps        nf= 1.00            nr= 1.00  
 ns= 1.00            ibe= 0. amps        ibc= 0. amps  
 subs= -1.00        expli= 1.0e+15 amps

2\*\*\* low current beta degradation effect parameters \*\*\*

isc= 5.7e-17 amps    ise= 1.1e-16 amps    nc= 1.77  
 ne= 1.52

3\*\*\* base width modulation parameters \*\*\*

vaf= 23.00 volts     var= 18.00 volts

4\*\*\* high current beta degradation effect parameters \*\*\*

ikf= 59.00u amps    nkf= 500.00m        ikr= 30.00u amps

5\*\*\* parasitic resistor parameters \*\*\*

irb= 1.20u amps     rb= 365.00 ohms     rbm= 25.00 ohms  
 re= 9.00 ohms       rc= 8.00 ohms       vo= 0. volts  
 gamma= 0.

6\*\*\* junction capacitor parameters \*\*\*

cbcp= 0. farad       cbep= 0. farad       ccsp= 0. farad  
 cjc= 190.00f farad    cje= 11.00f farad    cjs= 0. farad  
 fc= 950.00m        mjc= 430.00m        mje= 250.00m  
 mjs= 500.00m       vjc= 670.00m volts   vje= 650.00m volts  
 vjs= 750.00m volts   xcjc= 1.00            qco= 0. coul

7\*\*\* transit time parameters \*\*\*

itf= 550.00u amps      ptf= 9.00 deg k      tf= 16.00n secs  
 tr= 90.00n secs      vtf= 0.      xtf= 1.00

8\*\*\* temperature compensation parameters \*\*\*

tlev= 0.      tlevc= 0.      trel= 0. /deg  
 tre2= 0. /deg2      trb1= 0. /deg      trc1= 0. /deg  
 trb2= 0. /deg2      trm1= 0. /deg      xtb= 2.20  
 trm2= 0. /deg2      xti= 2.60      cte= 0. /deg  
 ctc= 0. /deg      cts= 0. /deg      trc2= 0. /deg2  
 tref= 25.00 deg c      bex= 2.42      bexv= 1.90

9\*\*\* noise parameters \*\*\*

kf= 0.      af= 100.00m

\*\*\*\*\*

\*\*\* bjt model parameters model name: 0:ps model type:pnv \*\*\*

\*\*\*\*\*

names	values	units	names	values	units	names	values	units
----	-----	-----	----	-----	-----	----	-----	-----

1\*\*\* basic dc parameters \*\*\*

level= 1.00      bf= 300.00      br= 1.00  
 brs= 0.      bulk= gnd      is= 20.00f amps  
 iss= 0. amps      nf= 1.00      nr= 1.00  
 ns= 1.00      ibe= 0. amps      ibc= 0. amps  
 subs= -1.00      expli= 1.0e+15 amps

2\*\*\* low current beta degradation effect parameters \*\*\*

isc= 0. amps      ise= 2.2e-16 amps      nc= 2.00

ne= 1.20

3\*\*\* base width modulation parameters \*\*\*

vaf= 36.00 volts      var= 12.00 volts

4\*\*\* high current beta degradation effect parameters \*\*\*

ikf= 13.00m amps      nkf= 500.00m      ikr= 1.30m amps

5\*\*\* parasitic resistor parameters \*\*\*

irb= 300.00u amps      rb= 270.00 ohms      rbm= 150.00 ohms

re= 10.00m ohms      rc= 30.00 ohms      vo= 0. volts

gamma= 0.

6\*\*\* junction capacitor parameters \*\*\*

cbcp= 0. farad      cbep= 0. farad      ccsp= 0. farad

cjc= 2.20p farad      cje= 1.00p farad      cjs= 0. farad

fc= 500.00m      mjc= 300.00m      mje= 430.00m

mjs= 500.00m      vjc= 500.00m volts      vje= 670.00m volts

vjs= 750.00m volts      xcjc= 1.00      qco= 0. coul

7\*\*\* transit time parameters \*\*\*

itf= 300.00m amps      ptf= 0. deg k      tf= 3.50n secs

tr= 500.00n secs      vtf= 0.      xtf= 1.00

8\*\*\* temperature compensation parameters \*\*\*

tlev= 0.      tlevc= 0.      trel= 0. /deg

tre2= 0. /deg2      trb1= 0. /deg      trc1= 0. /deg

trb2= 0. /deg2      trm1= 0. /deg      xtb= 2.20

trm2= 0. /deg2      xti= 3.00      cte= 0. /deg

ctc= 0. /deg      cts= 0. /deg      trc2= 0. /deg2

tref= 25.00 deg c      bex= 2.42      bexv= 1.90

9\*\*\* noise parameters \*\*\*

kf= 0.                af= 1.00

\*\*\*\*\*

\*\*\* bjt model parameters model name: 0:n5 model type:npn \*\*\*

\*\*\*\*\*

names	values	units	names	values	units	names	values	units
-----	-----	-----	-----	-----	-----	-----	-----	-----

1\*\*\* basic dc parameters \*\*\*

level= 1.00                bf= 130.00                br= 2.40  
 brs= 0.                bulk= gnd                is= 9.0e-16 amps  
 iss= 0. amps                nf= 1.00                nr= 1.00  
 ns= 1.00                ibe= 0. amps                ibc= 0. amps  
 subs= 1.00                expli= 1.0e+15 amps

2\*\*\* low current beta degradation effect parameters \*\*\*

isc= 0. amps                ise= 1.25f amps                nc= 2.00  
 ne= 1.60

3\*\*\* base width modulation parameters \*\*\*

vaf= 35.00 volts                var= 5.00 volts

4\*\*\* high current beta degradation effect parameters \*\*\*

ikf= 57.30m amps                nkf= 500.00m                ikr= 12.50m amps

5\*\*\* parasitic resistor parameters \*\*\*

irb= 750.00u amps                rb= 60.00 ohms                rbm= 13.00 ohms  
 re= 2.50 ohms                rc= 30.00 ohms                vo= 0. volts  
 gamma= 0.

6\*\*\* junction capacitor parameters \*\*\*

cbcp= 0. farad      cbep= 0. farad      ccsp= 0. farad  
 cjc= 660.00f farad      cje= 700.00f farad      cjs= 840.00f farad  
 fc= 500.00m      mjc= 380.00m      mje= 400.00m  
 mjs= 430.00m      vjc= 600.00m volts      vje= 850.00m volts  
 vjs= 670.00m volts      xcjc= 250.00m      qco= 0. coul

7\*\*\* transit time parameters \*\*\*

itf= 25.00m amps      ptf= 0. deg k      tf= 47.00p secs  
 tr= 18.00n secs      vtf= 0.      xtf= 1.00

8\*\*\* temperature compensation parameters \*\*\*

tlev= 0.      tlevc= 0.      trel= 0. /deg  
 tre2= 0. /deg2      trb1= 0. /deg      trc1= 0. /deg  
 trb2= 0. /deg2      trm1= 0. /deg      xtb= 2.50  
 trm2= 0. /deg2      xti= 2.90      cte= 0. /deg  
 ctc= 0. /deg      cts= 0. /deg      trc2= 0. /deg2  
 tref= 25.00 deg c      bex= 2.42      bexv= 1.90

9\*\*\* noise parameters \*\*\*

kf= 0.      af= 1.00

\*\*\*\*\* HSPICE -- H93A.01      11:04:15 95/05/08      sun4

\*\*\*\*\*

\*\*\*\*\* operating point information      tnom= 25.000 temp= 25.000

\*\*\*\*\*

\*\*\*\*\* operating point status is voltage simulation time is 0.

node =voltage      node =voltage      node =voltage

+0:1	=-907.4328u 0:2	= 7.5000 0:3	= -7.5000
+0:4	= 5.0000 0:5	= 0. 0:6	= 0.
+0:7	=-907.4328u 0:8	=-905.4327u 0:9	= 1.1344m
+0:10	=-905.4327u 0:11	= 1.1344m 0:12	= -1.4774
+0:13	= -2.0794 0:14	= 4.8152 0:15	= -2.8191
+0:16	= 4.0716 0:17	=-801.1297m 0:18	= 3.9292
+0:19	= -3.3527 0:20	= 3.9292 0:21	= -4.0233
+0:22	= -5.8580 0:23	= -7.1716 0:24	= -7.1705
+0:25	= -5.8527 0:26	= -5.0291 0:27	= -4.6938
+0:28	= -7.1705 0:29	= -2.0794 0:30	= -2.9317
+0:31	=-801.1297m 0:32	= -1.4774 0:33	= 4.8152
+0:34	= 4.0716 0:35	=-801.4285m 0:36	= -2.8235
+0:37	= -2.9317 0:38	=-825.3452m 0:39	=-825.3452m
+0:40	= -5.8527 0:41	= 3.0767 0:42	= 2.2876
+0:43	= 3.0767 0:44	= 2.2876 0:45	= -2.8235
+0:46	=-801.4285m 0:47	= -2.8191 0:48	= 0.
+0:49	= -2.9788m 0:50	= 4.1570 0:51	= 4.1570
+0:52	= -2.9788m 0:53	= 0. 0:54	= 0.
+0:55	= 0. 0:62	= 3.0713 0:63	= 2.5623
+0:64	= -5.9484 0:65	= 3.3408 0:66	= 3.3408
+0:67	= -3.7381 0:68	= -5.1428 0:69	= 3.0713
+0:70	= 2.6933 0:71	= 1.9372 0:72	= 2.6933
+0:73	= 2.6933 0:74	= -5.9729 0:75	= -5.9240
+0:76	= -6.9747 0:77	= 4.9255 0:78	= 4.9255
+0:79	= -6.4656 0:80	= -5.9328 0:81	= -5.9328
+0:82	= 2.6933 0:83	= 6.9058 0:84	= 3.5634
+0:85	= 3.4892 0:86	= 3.5634 0:87	= 3.4892
+0:88	= 2.8823 0:89	= 2.8823 0:90	= 6.9058
+0:95	=-757.5059m 0:96	= -5.9470 0:97	= 991.4472m
+0:98	= -5.1632 0:99	= -5.9966 0:100	= -5.8770
+0:105	=-757.5059m 0:106	= -5.9470 0:107	= 991.4472m

+0:108	= -5.1632	0:109	= -5.9966	0:110	= -5.8770
+0:116	= 7.1237	0:117	= 3.3389	0:118	= 2.2622
+0:119	= 7.4635	0:120	= -7.3713	0:121	= -7.4816
+0:122	= 7.4590	0:123	= -5.8606	0:124	= -408.8820m
+0:125	= -68.1470m	0:126	= -7.3535	0:127	= 1.7239
+0:128	= 4.1132	0:129	= 769.0119m	0:130	= -4.4377
+0:131	= -4.7780	0:132	= 6.7164	0:133	= 762.8025m
+0:134	= -7.1874	0:135	= -5.1508	0:136	= -7.2250
+0:137	= -355.2303m	0:138	= -7.2803	0:139	= -693.8484m
+0:140	= 1.4547	0:141	= 727.3420m	0:142	= -6.4910
+0:143	= -4.7780	0:144	= -7.4448	0:151	= 7.1237
+0:152	= 3.3389	0:153	= 2.2622	0:154	= 7.4635
+0:155	= -7.3713	0:156	= -7.4816	0:157	= 7.4590
+0:158	= -5.8606	0:159	= -408.8820m	0:160	= -68.1470m
+0:161	= -7.3535	0:162	= 1.7239	0:163	= 4.1132
+0:164	= 769.0119m	0:165	= -4.4377	0:166	= -4.7780
+0:167	= 6.7164	0:168	= 762.8025m	0:169	= -7.1874
+0:170	= -5.1508	0:171	= -7.2250	0:172	= -355.2303m
+0:173	= -7.2803	0:174	= -693.8484m	0:175	= 1.4547
+0:176	= 727.3420m	0:177	= -6.4910	0:178	= -4.7780
+0:179	= -7.4448	0:186	= 779.3668m	0:187	= 6.7127
+0:188	= 2.2227	0:189	= -4.7879	0:190	= -4.4489
+0:191	= -694.2499m	0:192	= -5.1619	0:193	= -356.8681m
+0:194	= -7.2256	0:195	= -7.3053	0:196	= -7.1880
+0:197	= 7.4617	0:198	= -5.8497	0:199	= -7.3715
+0:200	= 3.3443	0:201	= -68.0194m	0:202	= 7.1246
+0:203	= -408.1162m	0:204	= 4.1211	0:205	= -7.4816
+0:206	= 2.2654	0:207	= 7.4571	0:208	= 1.7260
+0:209	= -7.4027	0:210	= 1.4562	0:211	= 728.1238m
+0:212	= -6.4917	0:213	= -4.7879	0:214	= -7.4449
+0:221	= 779.3582m	0:222	= 6.7127	0:223	= 2.2227

+0:224 = -4.7879 0:225 = -4.4489 0:226 = -694.2490m  
 +0:227 = -5.1619 0:228 = -356.8672m 0:229 = -7.2256  
 +0:230 = -7.3053 0:231 = -7.1880 0:232 = 7.4617  
 +0:233 = -5.8497 0:234 = -7.3715 0:235 = 3.3443  
 +0:236 = -68.0194m 0:237 = 7.1246 0:238 = -408.1162m  
 +0:239 = 4.1211 0:240 = -7.4816 0:241 = 2.2654  
 +0:242 = 7.4571 0:243 = 1.7260 0:244 = -7.4027  
 +0:245 = 1.4562 0:246 = 728.1238m 0:247 = -6.4917  
 +0:248 = -4.7879 0:249 = -7.4449 0:256 = 779.3582m  
 +0:257 = 6.7127 0:258 = 2.2227 0:259 = -4.7879  
 +0:260 = -4.4489 0:261 = -694.2490m 0:262 = -5.1619  
 +0:263 = -356.8672m 0:264 = -7.2256 0:265 = -7.3053  
 +0:266 = -7.1880 0:267 = 7.4617 0:268 = -5.8497  
 +0:269 = -7.3715 0:270 = 3.3443 0:271 = -68.0194m  
 +0:272 = 7.1246 0:273 = -408.1162m 0:274 = 4.1211  
 +0:275 = -7.4816 0:276 = 2.2654 0:277 = 7.4571  
 +0:278 = 1.7260 0:279 = -7.4027 0:280 = 1.4562  
 +0:281 = 728.1238m 0:282 = -6.4917 0:283 = -4.7879  
 +0:284 = -7.4449 0:291 = 779.3668m 0:292 = 6.7127  
 +0:293 = 2.2227 0:294 = -4.7879 0:295 = -4.4489  
 +0:296 = -694.2499m 0:297 = -5.1619 0:298 = -356.8681m  
 +0:299 = -7.2256 0:300 = -7.3053 0:301 = -7.1880  
 +0:302 = 7.4617 0:303 = -5.8497 0:304 = -7.3715  
 +0:305 = 3.3443 0:306 = -68.0194m 0:307 = 7.1246  
 +0:308 = -408.1162m 0:309 = 4.1211 0:310 = -7.4816  
 +0:311 = 2.2654 0:312 = 7.4571 0:313 = 1.7260  
 +0:314 = -7.4027 0:315 = 1.4562 0:316 = 728.1238m  
 +0:317 = -6.4917 0:318 = -4.7879 0:319 = -7.4449

Opening plot unit= 15

file= ./listapot.tr0

opening plot unit= 9

file= ./listapot.mt0

\*\*\*\*\*

\*\*\*\*\* transient analysis            tnom= 25.000 temp= 25.000

\*\*\*\*\*

idd            = 8.2983E-02 from= 0.0000E+00    to= 3.0000E-06

\*\*\*\*\* job concluded

\*\*\*\*\* HSPICE -- H93A.01        11:04:15 95/05/08    sun4

\*\*\*\*\*

\*\*\*\*\* job statistics summary        tnom= 25.000 temp= 25.000

\*\*\*\*\*

total memory used        718 kbytes

# nodes = 727 # elements= 367

# diodes= 0 # bjts = 152 # jfets = 0 # mosfets = 0

analysis	time	# points	tot. iter	conv.iter
----------	------	----------	-----------	-----------

op point	2.05	1	20	
----------	------	---	----	--

transient	404.02	3001	4026	2013 rev= 0
-----------	--------	------	------	-------------

readin	2.55			
--------	------	--	--	--

errchk	0.53			
--------	------	--	--	--

---

```
setup      0.70
output     0.07
total cpu time    409.90 seconds
job started at 11:04:15 95/05/08
job ended  at 11:11:33 95/05/08
```

lic: Releasing license for hspice

```
>info:      ***** hspice job concluded
          440.4 real    403.7 user    6.6 sys
```

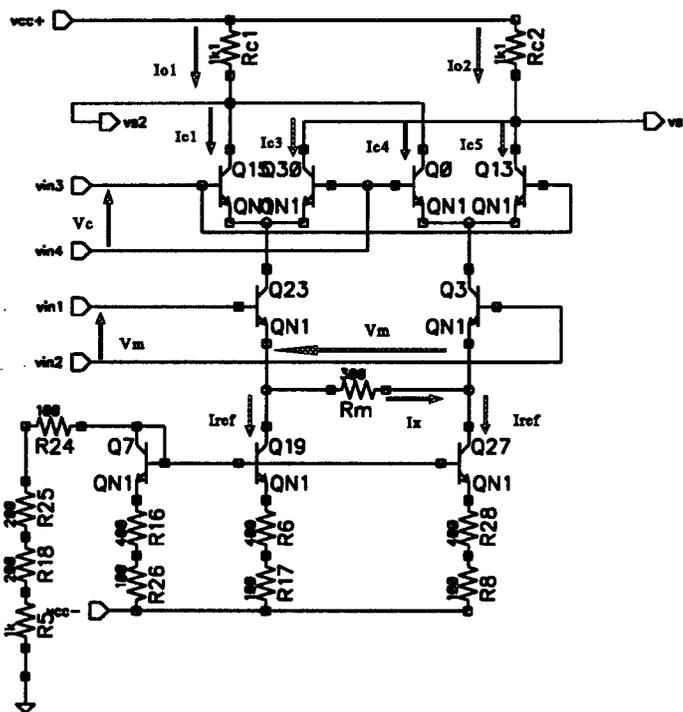
I am done with listapot

Mon May 8 11:11:34 GMT 1995

## Apéndice IV

El contenido de este Apéndice corresponde al desarrollo matemático de la demostración del funcionamiento del multiplicador.

Para todos los cálculos vamos a suponer que la  $V_{bc}$  es de 0,7 Voltios.



$$I_{c2} \cdot \left( I_{ref} - \frac{V_m}{R_m} \right) \left( \frac{1}{1 - e^{-\frac{V_c}{V_T}}} \right)$$

$$I_{c4} = \left( I_{ref} \cdot \frac{V_m}{R_m} \right) \left( \frac{1}{1 \cdot e^{-\frac{V_c}{V_t}}} \right)$$

$$I_{c3} = \left( I_{ref} \cdot \frac{V_m}{R_m} \right) \left( \frac{1}{1 \cdot e^{-\frac{V_c}{V_t}}} \right)$$

$$I_{c1} = \left( I_{ref} \cdot \frac{V_m}{R_m} \right) \left( \frac{1}{1 \cdot e^{-\frac{V_c}{V_t}}} \right)$$

Podemos observar que  $I_{o1} = I_{c1} + I_{c4}$ , y por lo tanto:

$$I_{o1} = I_{ref} \cdot \frac{V_m}{R_m} \left( \frac{e^{\frac{V_c}{V_t}} - 1}{e^{\frac{V_c}{V_t}} + 1} \right)$$

La  $I_{o2} = I_{c2} + I_{c3}$  quedará:

$$I_{o2} = I_{ref} \cdot \frac{V_m}{R_m} \left( \frac{1 - e^{-\frac{V_c}{V_t}}}{e^{-\frac{V_c}{V_t}} + 1} \right)$$

Como se puede observar en la figura  $V_{o1} = V_{cc} - I_{o1}R_c$  y  $V_{o2} = V_{cc} - I_{o2}R_c$ . Por lo tanto:

$$V_0 = (V_{o1} - V_{o2}) = R_c(I_{o2} - I_{o1})$$

$$V_0 = R_c \frac{V_m}{R_m} \frac{\left( e^{-\frac{V_c}{2V_t}} - e^{\frac{V_c}{2V_t}} \right)}{\left( e^{-\frac{V_c}{2V_t}} + e^{\frac{V_c}{2V_t}} \right)}$$

Utilizando la definición de Tangente Hiperbólica quede de la siguiente forma:

$$V_0 = -2 \frac{V_m}{R_m} R_c \tanh \left( \frac{V_c}{2Vt} \right)$$

Como la tangente hiperbólica es :  $\frac{e^x - e^{-x}}{e^x + e^{-x}}$

- Cuando x tiende a infinito es 1.
- Cuando x tiende a menos infinito es -1.

Dibujando la función observamos que en el margen de 0,7 a -0,7 la función es lineal y la  $\tanh(x)$  es aproximadamente igual a x. Por esta razón podremos expresar la salida como:

$$V_0 = -2 \frac{V_m}{R_m} R_c \left( \frac{V_c}{2Vt} \right)$$

siempre que el cociente de las tensiones está entre los márgenes indicados.

$$V_0 = -K V_m V_c$$

Donde K es la ganancia.

# Bibliografía

1.- Bipolar and Mos Analog Circuits Design.

Autor: Alan G. Grebenne.

Editorial: Willey.

2.- Electrónica Integrada.

Autor: Millmann and Halkias.

Editorial: Hispano Europea.

3.- Analog Integrated Circuits.

Autor: Gray Meyer.

Editorial: Wiley.

4.- Circuitos Electrónicos.

Autores: Donald L. Schilling.

Charles Belove.

Editorial:

5.- Analog Electronics Handbook.

Autores: T.H. Collins.

Editorial: Prentice Hall.

6.- Spice.

Autor: E. Muñoz.

Editorial: ETSIT Madrid.

7.- Design of MOS VLSI Circuits for Telecommunications.

Autor: Tsividis Antognetti.

Editorial: Prentice Hall.

**8.- Basic VLSI Design.**

Autores: Preknell - Eshraghiam

Editorial: Prentice Hall

**9.- Electronics Devices. Circuits and Aplications.**

Autor: T. H. Davis

Editorial: Prentice Hall.

**10.- Diseño de Circuitos con Transistores.**

Autor: D. T. Horn.

Editorial: Paraninfo.

**11- Computers Hided Analysis of Active Circuits.**

Autor: ...

Editorial: Dekker.

**12.- Revista Mundo Electrónico.**

Nº 201, 202, 213

Editorial: Boixareu Editores, S.A.

**13.- Spice.**

Autor: Tuinenga

Editorial: Prentice Hall

**14.- CMOS Analog Circuit Design.**

Autor: Allan Holberg

Editorial: HRW

**15.- Verification and Simulation Tools.**

Editorial: OrCAD System Corporation.

# Planos

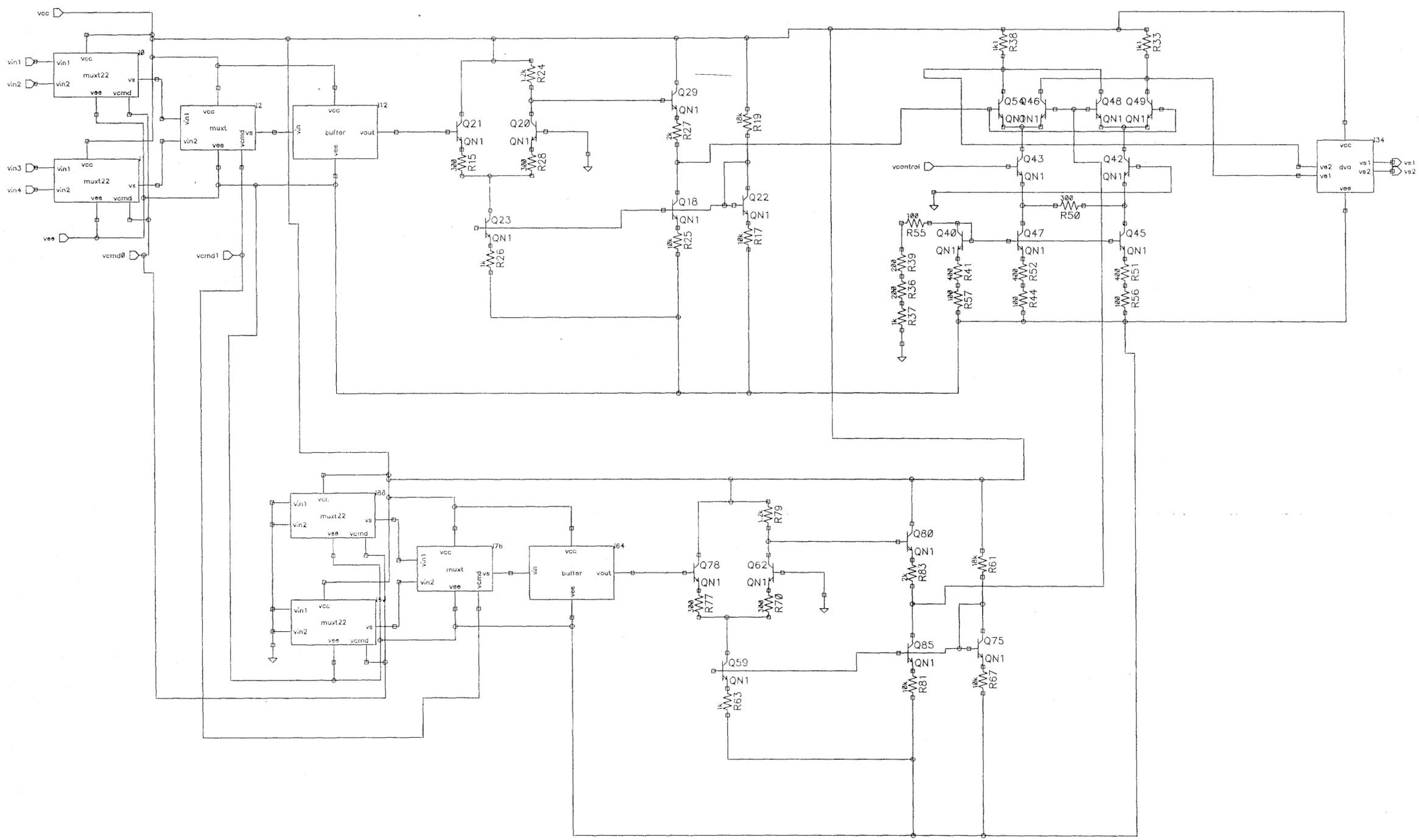
4

3

2

1

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPROVED



4

3

2

1

APPROVALS		DATE
DRAWN	FMM	
CHECKED		
CHECKED		
CHECKED		
ISSUED		

Matriz_Conmutacion Control_de_Ganancia		SIZE	FSCM NO.	DWG NO.	REV
		C			
SCALE	SHEET		OF		

# MATRIZ\_DE\_CONMUTACION\_4\_a\_1\_CONTROL\_DE\_GANANCIA

