

# Capítulo 4

## Pruebas.

Para comprobar el éxito de la portabilidad del conjunto de herramientas ALLIANCE, se ejecuta y se comparan los resultados en el sistema original y en el sistema portado, uno de los tutoriales más completos que viene con la herramienta ALLIANCE, el cual es la implementación del circuito integrado AMD2901.

### 4.1 Tutorial: AMD 2901.

La intención de este tutorial es presentar el flujo de diseño típico de un procesador simple de 4 bits (ADM2901), usando el conjunto de herramientas ALLIANCE.

Las herramientas usadas aquí son:

- **asimut:** compilador y simulador VHDL.
- **genlib:** captura del netlist.
- **scr:** localizador y ruteador de celdas estándares.
- **ring:** localizador y ruteador desde el núcleo del *chip* a los *pads* que conforman el anillo exterior al núcleo del *chip*.
- **lynx:** extractor de layouts simbólicos.
- **lvx:** comparador de netlist.
- **druc:** verificador de reglas de diseño.
- **graal:** editor gráfico de layout simbólico (solo para ambiente UNIX).
- **desb:** abstractor funcional.
- **proof:** herramienta de pruebas formales entre dos descripciones de comportamiento.
- **s2r:** convertidor del layout simbólico a layout real.

El flujo de diseño usado en este tutorial esta compuesto de 5 pasos:

- **Paso 1:** descripción y simulación del comportamiento (modelo VHDL).
- **Paso 2:** generación y validación de la descripción estructural (compuertas netlist).

- **Paso 3:** layout del diseño físico (localización y ruteado).
- **Paso 4:** extracción y verificación.
- **Paso 5:** finalización del *chip*.

Todos los archivos fuentes necesarios son suministrados con este tutorial:

- **amd.vbe:** modelo de comportamiento VHDL.
- **pattern.pat:** patrones de simulación.
- **chip.c:** descripción estructural del *chip*.
- **chip.rin:** localización de los *pads* (para RING).
- **chip.inf:** transformación de nombres de registros (para DESB).

Se usan el conjunto de herramientas ALLIANCE (las originales y las portadas) para validar estos archivos de entrada y para construir el layout físico. La salida es un archivo layout CIF listo para la fabricación del *chip*.

Todas las herramientas ALLIANCE usan un conjunto de variables de ambiente. Estas variables son asignadas por el comando “*export*” para el sistema operativo UNIX y por el comando “*set*” para el sistema operativo WINDOWS 95.

Algunas variables de ambiente indican una lista de rutas absolutas de archivos o librerías.

Para UNIX:

```
MBK_WORK_LIB=.
MBK_CATA_LIB=./alliance/cells/scr:/alliance/cells/ring
export MBK_WORK_LIB MBK_CATA_LIB
```

Para Windows 95:

```
set MBK_WORK_LIB=.
set MBK_CATA_LIB=./alliance/cells/scr:/alliance/cells/ring
```

La variable de ambiente MBK\_WORK\_LIB define el directorio de trabajo. Todos los archivos de usuarios serán escritos en este directorio.

La variable de ambiente MBK\_CATA\_LIB define todas las rutas de los directorios que contiene las librerías de celdas predefinidas en el ALLIANCE.

## 4.1.1 Especificación del comportamiento.

### 4.1.1.1 Modelo de comportamiento.

El comportamiento del circuito esta descrito en el archivo *amd.vbe* usando el subconjunto **ALLIANCE VHDL**. Se puede correr el compilador **VHDL** para validar la sintaxis del archivo **VHDL**.

```
>asimut -b -c amd
```

- **amd** : es el nombre del archivo que contiene la descripción del comportamiento (*amd.vbe*).
- **-b** : significa que el archivo es una descripción de comportamiento pura.
- **-c** : corre la herramienta solamente para compilación (se verifica la sintaxis, pero no se desarrolla la simulación).

### 4.1.1.2 Patrones de simulación.

Así como se define una especificación formal, se define un conjunto de patrones de simulación. Estos mismos patrones se pueden usar para validar cada paso del diseño desde la especificación hasta el layout físico. El archivo *pattern.pat* se proporciona como ejemplo.

### 4.1.1.3 Simulación.

Se tiene ahora una descripción lógica del circuito, y una lista de patrones. Se puede correr el simulador VHDL ASIMUT. Se debe asignar algunas variables de ambientes específicas para ASIMUT.

Para UNIX:

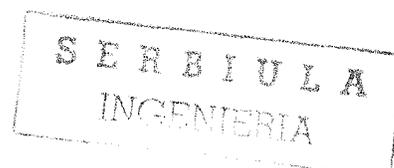
```
VH_MAXERR=10
VH_PATSFY=pat
export VH_MAXERR VH_PATSFY
```

Para Windows 95:

```
set VH_MAXERR=10
set VH_PATSFY=pat
```

La variable de ambiente **VH\_MAXERR** indica el máximo número de errores aceptados antes que ASIMUT detenga la simulación.

La variable de ambiente **VH\_PATSFY** define la extensión del archivo de patrón de simulación.



### >asimut -b amd pattern result\_beh

- **amd** es el nombre del archivo que contiene la descripción del comportamiento (*amd.vbe*).
- **pattern** es el nombre del archivo que contiene los patrones de entrada (*pattern.pat*).
- **result\_beh** es el nombre del archivo para los patrones resultantes (*result\_beh.pat*).
- **-b** significa que el archivo *amd* es una descripción de comportamiento pura.

En el archivo de patrones de entrada se pueden especificar salidas esperadas (predecidas). Cualquier diferencia entre las salida predecidas en *pattern.pat* y los resultados de la simulación serán reportados como comentarios en la pantalla y en el archivo *result\_beh.pat*. Esta salida puede ser usada nuevamente para otra simulación, ya que tiene el mismo formato que el archivo de entrada.

## 4.1.2 Descripción estructural.

### 4.1.2.1 Diseño estructural.

El objetivo de este paso es construir el esquema a nivel de compuertas correspondiente a la especificación del comportamiento.

Se debe instanciar y conectar las compuertas lógicas y los *pads*, suministradas en la librería de celdas estándares SCR y la librería de *pads* RING. Para esto, ALLIANCE usa el lenguaje de procedimientos GENLIB, el cual es un conjunto de funciones en C.

La descripción estructural del AMD2901 sigue un enfoque jerárquico:

- Primero, cada bloque funcional está descrito como un netlist de compuertas separado. Hay 5 bloques: *accu*, *alu*, *ram*, *muxe*, *muxout*.
- El próximo nivel jerárquico es el "heart" que interconecta estos cinco bloques entre si.
- El último nivel jerárquico es el "chip" que conecta el "heart" con los "pads".

El archivo *chip.c* usa el lenguaje GENLIB para describir estos siete bloques jerárquicos.

El sistema ALLIANCE acepta algunos formatos externos de archivos para el netlist (EDIF, VHDL, SPICE, ALLIANCE, COMPASS). Las siguientes variables de ambiente asignan el formato. La extensión **.vst** corresponde al VHDL estructural.

Para UNIX:

```
MBK_IN_LO=vst
MBK_OUT_LO=vst
export MBK_IN_LO MBK_OUT_LO
```

Para Windows 95:

```
set MBK_IN_LO=vst
set MBK_OUT_LO=vst
```

La variable de ambiente MBK\_IN\_LO define el formato del netlist de entrada.

La variable de ambiente MBK\_OUT\_LO define el formato del netlist de salida.

Todos los archivos netlist creados y usados por GENLIB serán VHDL estructurales.

**>genlib chip**

De esta manera se generan los siete archivos (*alu.vst*, *accu.vst*, *ram.vst*, *muxe.vst*, *muxout.vst*, *heart.vst*, *chip.vst*) del enfoque jerárquico de la descripción estructural del AMD2901.

#### 4.1.2.2 Simulación del modelo estructural.

Los diferentes archivos netlist que se han creado representan la descripción estructural que debe ser validada por simulación, usando los mismos patrones como en el paso 1.

**>asimut chip pattern result\_str**

- **chip** es el archivo estructural de entrada (*chip.vst*).
- **pattern** es el nombre del archivo que contiene los patrones de entrada (*pattern.pat*).
- **result\_str** es el nombre del archivo para los patrones resultantes (*result\_str.pat*).

Note que la opción **-b** ya no aparece, ya que ahora tenemos un modelo estructural.

Cualquier error que se produzca será reportado en el archivo *result\_str.pat*.

### 4.1.3 Diseño físico.

La construcción del *heart* y el ruteado del *heart* a los *pads* son tareas diferentes y delicadas, por lo tanto se emplean diferentes herramientas.

#### 4.1.3.1 Ruteado del heart.

SCR es una herramienta de localización y ruteado de celdas estándares.

Aquí, ALLIANCE nuevamente acepta algunos formatos externos de archivos para el layout simbólico (COMPASS, ALLIANCE, MODGEN). Las siguientes variables de ambiente se emplean para indicar los formatos a ser utilizados.

Para UNIX:

```
MBK_IN_PH=ap
MBK_OUT_PH=ap
export MBK_IN_PH MBK_OUT_PH
```

Para Windows 95:

```
set MBK_IN_PH=ap
set MBK_OUT_PH=ap
```

La variable de ambiente MBK\_IN\_PH indica el formato del layout de entrada.

La variable de ambiente MBK\_OUT\_PH indica el formato del layout de salida.

Antes de realizar el ruteado del netlist, la herramienta SCR convierte este netlist jerárquico en un netlist no jerárquico (el netlist es linealizado).

```
>scr -p -r -i 1000 heart
```

- **heart** es el netlist de entrada (*heart.vst*). El layout de salida será *heart.ap*.
- **-p** : realiza la localización automática.
- **-r** : requiere ruteado.
- **-i 1000** : número de iteraciones (incrementando este número mejorará la localización).

De esta manera se genera el archivo layout simbólico *heart.ap*.

#### 4.1.3.2 Verificación del heart.

- Chequeo de las reglas del layout simbólico.

El chequeo de las reglas del layout simbólico es realizado por la herramienta DRUC.

```
>druc heart
```

Si es necesario, DRUC genera un archivo de error (*core.err*).

- Extracción del netlist.

LYNX es un extractor jerárquico que provee un netlist de compuertas.

Para evitar colisiones de nombres, se usa otro formato de archivo para el netlist extraído (formato *.ai*). Este nuevo formato se usa para data adicional: "parásitos de capacitancia extraídos".

Para UNIX:

```
MBK_OUT_LO=ai
export MBK_OUT_LO
```

Para Windows 95:

```
set MBK_OUT_LO=ai
```

La variable de ambiente MBK\_OUT\_LO indica el formato del netlist de salida.

Entonces se corre el extractor LYNX.

```
>lynx heart
```

De esta manera se genera el archivo netlist extraído *heart.ai*.

- Comparación del netlist.

La próxima herramienta es el comparador de netlist LVX, el cual desarrolla una comparación entre el netlist de entrada *heart.vst* y el netlist extraído *heart.ai*, luego de hacer una linealización hasta el nivel de compuertas.

```
>lvx vst ai heart heart -f
```

### 4.1.3.3 Ruteado del chip.

La localización de los *pads* depende de coacciones externas. El archivo *chip.rin* define las coacciones de localización de los *pads*.

**>ring chip chip**

- **chip**: los archivos de entrada *chip.vst* y *chip.rin* (se debe usar el mismo nombre *chip* para los dos archivos).
- **chip**: el archivo de salida físico *chip.ap*.

De esta manera se genera el layout simbólico *chip.ap*.

En este punto, se puede visualizar el *chip* con el editor simbólico GRAAL (solo en el entorno UNIX).

Se puede ver el *chip* y el *heart* instanciado, usando el comando GRAAL para ir a través de los niveles jerárquicos.

**>graal**

## 4.1.4 Validación física.

Las herramientas de verificación ALLIANCE realizan tanto la verificación jerárquica como la verificación lineal (no jerárquica). Aquí se empleará un enfoque jerárquico.

### 4.1.4.1 Verificación del chip.

El mismo procedimiento usado para el *heart* lo se aplicará para el *chip*.

**>druc chip**

Eventualmente, DRUC generará un archivo de error (*chip.err*).

**>lynx chip**

De esta manera se genera el archivo de netlist extraído *chip.al* (este netlist instancia el *heart*).

**>ivx vst al chip chip**

Ambos netlist *chip.vst* y *chip.ai* son linealizados (no jerárquicos) al nivel de compuertas por LVX antes de la comparación.

#### 4.1.4.2 Simulación del chip.

Finalmente se puede chequear globalmente el netlist extraído aplicándole los patrones originales. Gracias al enfoque multiformato, el simulador ASIMUT acepta el formato *.ai* como descripción estructural de entrada.

```
Para UNIX:
    MBK_IN_LO=ai
    export MBK_IN_LO
```

```
Para Windows 95:
    set MBK_IN_LO=ai
```

La variable de ambiente MBK\_IN\_LO indica en formato del netlist de entrada.

Entonces se debe correr el simulador ASIMUT.

```
>asimut chip pattern res_pattern
```

Para completar la validación del *chip*, se realizará la abstracción funcional y seguidamente las pruebas.

#### 4.1.4.3 Abstracción funcional y prueba formal.

La abstracción funcional del netlist de transistores es realizada por la herramienta DESB. DESB linealiza el chip (*chip.ai*) para obtener la descripción a nivel de transistores, y se abstrae entonces una descripción del comportamiento (*chip.vbe*). Para mantener la coherencia de los nombres de registros entre la descripción del comportamiento dada por DESB (*chip.vbe*) y la descripción del comportamiento inicial (*amd.vbe*), se requiere un archivo describiendo la transformación de nombres. Este archivo es *chip.inf*.

```
>desb chip -i -v
```

- **chip**: es el layout de entrada (*chip.ai*).
- **-i**: lee el archivo *chip.inf*.
- **-v**: para vectorizar la interfaz de la descripción del comportamiento.

De esta manera se genera la descripción del FLUJO DE DATOS VHDL: *chip.vbe*.

Ahora se tiene la descripción del comportamiento abstraída (*chip.vbe*) y la descripción del comportamiento inicial (*amd.vbe*).

Con PROOF se puede chequear la equivalencia formal entre las dos descripciones.

**>proof -d amd chip**

- **chip**: es la descripción del comportamiento abstraída (*chip.vbe*).
- **amd**: es la descripción del comportamiento inicial (*amd.vbe*).
- **-d**: muestra los errores.

### 4.1.5 Finalización del chip.

Hasta ahora se ha usado layout simbólico (todas las coordenadas en unidades lambda). Se debe ahora convertirlas a dimensiones reales, creando el layout físico (se soportan dos formatos de salida: CIF y GDSII). Este último paso lo realiza S2R. S2R realiza la expansión de simbólica a real, el "llenado de huecos", y "marcación e invocación de celdas físicas preexistentes" (esto es necesario para los *pads*).

Se debe definir tres nuevas variables de ambiente:

Para UNIX:

```
RDS_TECHNO_NAME=/alliance/etc/pro12.rds
RDS_IN=cif
RDS_OUT=cif
export RDS_TECHNO_NAME RDS_IN RDS_OUT
```

Para Windows 95:

```
set RDS_TECHNO_NAME=/alliance/etc/pro12.rds
set RDS_IN=cif
set RDS_OUT=cif
```

La variable de ambiente RDS\_TECHNO\_NAME define el nombre del archivo de tecnología que contiene los parámetros correspondientes para el proceso destino (en este caso el proceso destino es 1.2 micron).

La variable de ambiente RDS\_IN define el formato de entrada para el layout de celdas preexistentes (*pads*).

La variable de ambiente RDS\_OUT define el formato de salida.

**>s2r -c chip amd2901**

- **-c**: elimina los conectores en el más alto nivel de jerarquía.

El archivo producido es **amd2901.cif**, el cual está listo para la fabricación.

bdigital.ula.ve

# Capítulo 5

## Conclusiones y Recomendaciones.

### 5.1 Conclusiones.

El sistema CAD ALLIANCE es un conjunto completo de herramientas CAD y librerías portables para la investigación y educación en diseño digital VLSI, que ha sido desarrollado por el Laboratorio MASI de la Universidad Pierre et Marie Curie en París, Francia. Es una herramienta de dominio público y totalmente gratis bajo los términos de la Licencia Pública General GNU.

El sistema CAD ALLIANCE es una herramienta CAD de diseño microelectrónico:

- *Completa*: porque permite abordar el diseño completo de un chip, desde la concepción en la mente de un ingeniero hasta la creación final de las máscaras.
- *Compleja*: en cuanto permite abordar diseño de circuitos muy complejos (diseños de hasta 800.000 transistores).
- *Fácil de utilizar*: a pesar de la complejidad intrínseca de una herramienta de este tipo. Las herramientas, comandos y demás que se manejan, son de fácil manejo.
- *Requiere relativamente poca infraestructura*: las herramientas similares a ALLIANCE suelen correr en estaciones de trabajo, en cambio ALLIANCE, a través del sistema operativo Linux, puede correr en un PC.

Con estas conclusiones, se puede afirmar que la herramienta ALLIANCE es especialmente adecuada para la enseñanza de la microelectrónica.

La migración del sistema CAD ALLIANCE al sistema operativo WINDOWS 95, permite que la herramienta esté disponible ampliamente para un gran número de estudiantes y profesores que están inmersos en el desarrollo de esta moderna tecnología de diseño que está siendo adoptada por diferentes universidades en el mundo y, ahora por nuestra ilustre Universidad de Los Andes.

La migración del sistema CAD ALLIANCE al sistema operativo WINDOWS 95 se logró mediante la emulación de un entorno UNIX en este sistema operativo, empleando para ello un conjunto de herramientas de desarrollo de software para

diferentes ambientes y sistemas operativos llamado *GNUPro™ Toolkit* de la compañía CYGNUS Solutions. El *GNUPro™ Toolkit* ofrece productividad, flexibilidad, desempeño y portabilidad con sus compiladores ANSI C y ANSI C++, aplicaciones utilitarias y librerías.

Con la emulación del entorno UNIX en el sistema operativo WINDOWS 95, se logró que el conjunto de herramientas ALLIANCE corra en el sistema operativo WINDOWS 95 conservando todas sus funcionalidades, es decir, el uso de las herramientas ALLIANCE en WINDOWS 95 es similar al uso que tienen estas herramientas en las entregas para los sistemas operativos UNIX, permitiendo que el trabajo del diseñador sea independiente del sistema operativo en el cual se ejecutan las herramientas e intercambiar los resultados de las diferentes fases de diseño entre un sistema operativo y otro.

## **5.1 Recomendaciones.**

Debido a que el entorno UNIX emulado en el sistema operativo WINDOWS 95 no posee las capacidades gráficas del entorno UNIX, las herramientas gráficas del sistema CAD ALLIANCE no pudieron ser emigradas, por lo que se recomienda realizar una investigación sobre la implementación de los entornos gráficos de sistemas operativos UNIX bajo WINDOWS 95 para luego emigrar estas herramientas.

# Anexos

## A.1 GNU Licencia Pública General.

### A.1.1 GNU Licencia Pública General. Versión 2, Junio 1991.

Copyright (C) 1989, 1991 Free Software Foundation, Inc.  
675 Mass Ave, Cambridge, MA 02139, USA

Se permite a cualquier persona copiar y distribuir copias textuales de esta licencia, pero no se permiten cambios.

#### Preámbulo

Las licencias para la mayoría de los software se diseñan para eliminar la libertad de compartirlo y cambiarlo. Por el contrario, la Licencia Pública General GNU, intenta garantizar la libertad de compartir y hacer cambios en el software gratuito (para asegurarse que el software es gratuito para todos sus usuarios). Esta Licencia Pública General se aplica a la mayoría de los software de la Fundación del Software Gratis y a cualquier otro programa cuyos autores concedan su uso. (Algún otro software de la Fundación de Software Gratis es cubierto por la Licencia Pública General de Librería GNU). Usted puede aplicarlo también a sus programas.

Cuando hablamos de software gratuito, nos referimos a libertad, no al precio. Nuestras Licencias Públicas se diseñan para asegurar que usted tenga la libertad de distribuir copias de software gratuito (y cobrar por ese servicio si así lo desea), que reciba los fuentes del código, o pueda conseguirlos si lo quiere, que pueda cambiar el software o usar parte de él en nuevos programas gratuitos, y que usted sepa que puede hacer estas cosas.

Para proteger sus derechos, necesitamos colocar restricciones que le prohíban a otros negarle estos derechos o pedirle que renuncie a ellos. Estas restricciones trasladan ciertas responsabilidades a usted si distribuye copias del software o si lo modifica.

Por ejemplo si usted distribuye copias de un programa, sea gratis o no, debe dar al comprador todos los derechos que usted tiene. Debe asegurarse de que ellos también reciban o puedan obtener el código fuente. Y usted debe hacerles saber estos términos para que conozcan sus derechos.

Protegemos nuestros derechos en dos pasos: 1) haciendo el Copyright del software y 2) ofreciendo esta licencia que le da permiso legal para copiar, distribuir, y/o modificar el software.

Además, para protección de cada autor y nuestra, queremos asegurar que cada uno comprenda que no hay garantía para este software gratis. Si el software es modificado por alguna persona y esta lo distribuye, queremos que quien lo adquiere sepa que lo que tiene no es el original, así que cualquier problema introducido por otros no incidirán en la reputación del autor original.

Por último, cualquier programa gratis esta amenazado constantemente por las patentes de software. Deseamos evitar el peligro de que los redistribuidores de un programa gratis obtengan licencias patentadas, haciéndose propietarios de los programas. Para prevenir esto, hemos aclarado que cualquier patente debe tener licencia para su uso gratis por cualquier persona o no tener licencia alguna.

Los términos precisos y condiciones para copiar, distribuir y modificar se enuncian a continuación.

### **A.1.2 Términos y condiciones para el copiado, distribución y modificación.**

0. Esta licencia se aplica a cualquier programa u otro trabajo que contenga alguna nota del dueño del copyright en la que diga que puede ser distribuido bajo los términos de esta Licencia Publica General. El "Programa", abajo, se refiere a cualquier programa o trabajo, y un "trabajo basado en el Programa" significa el programa o cualquier trabajo derivado bajo la ley del copyright: quiere decir, un trabajo que contenga el Programa o una porción de el ,sea una copia fiel o con modificaciones y/o traducido a cualquier otro lenguaje. (De aquí en adelante, traducción se incluye sin limite en el término modificación). Cada poseedor de la licencia se nombrara como "Usted".

Actividades diferentes del copiado, distribución y modificación no están cubiertas por esta licencia; están fuera de su alcance. El acto de ejecutar el Programa no está restringido, y la salida del Programa está cubierta solo si sus contenidos constituyen un trabajo basado en el Programa (independiente o

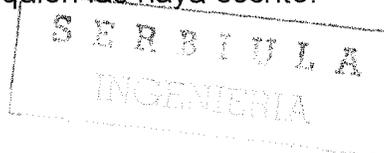
habiendo sido hecho con la ejecución del programa). Si esto es cierto depende en lo que haga el Programa.

1. Usted puede copiar y distribuir copias fieles del código fuente del Programa tal como lo recibe, en cualquier medio, siempre que usted publique en cada copia una nota apropiada y visible del copyright y abandono de la garantía; y dar a los otros usuarios del Programa una copia de esta licencia junto con el mismo.

Usted puede cobrar una cantidad por el acto físico de transferir una copia, y puede ofrecer protección de garantía a cambio de una gratificación si así lo desea.

2. Usted puede modificar su copia o copias del Programa o cualquier parte de él, formando así un trabajo basado en el Programa, y copiar o distribuir tales modificaciones o trabajo bajos los términos de la sección 1, siempre que usted cumpla con estas condiciones:
  - a. Usted debe hacer que los archivos modificados lleven notas donde manifieste que se cambiaron los archivos y la fecha de cualquier cambio.
  - b. Usted debe hacer que cualquier trabajo que usted distribuya o publique, que en total o en parte contiene o se deriva del Programa o cualquier parte de él, tenga licencia como un todo sin cargo para las terceras partes bajo los términos de esta Licencia.
  - c. Si el programa modificado normalmente lee comandos interactivamente al ejecutarse, usted debe hacer que al comenzar a ejecutarse para tales comandos comunes, se imprima o mostrar un aviso acerca del copyright y la no garantía (o de lo contrario diciendo que usted ofrece una garantía) y que los usuarios puedan redistribuir el programa bajo estas condiciones, y decirles como ver una copia de esta licencia. (Excepción: si el Programa es interactivo pero no imprime tales avisos, su trabajo basado en el Programa no tiene porque mostrar tales avisos).

Estos requerimientos se aplican para el trabajo modificado como una totalidad. Si secciones identificables de ese trabajo no se derivan del Programa y pueden razonablemente ser considerados trabajos separados e independientes, entonces esta licencia y sus términos no aplican para esas secciones cuando usted las distribuya como trabajos separados. Pero cuando usted distribuya las mismas secciones como partes de un todo que es un trabajo basado en el programa, la distribución del todo debe ser bajo los términos de esta licencia, cuyos permisos a otras personas se extienden en su totalidad, y así para cada una de las partes independientemente de quien las haya escrito.



Por lo tanto no es la intención de esta sección reclamar los derechos o disputarlos a trabajos escritos enteramente por usted; al contrario, la intención es ejercer el derecho de controlar la distribución de trabajos derivados o colectivos basados en el Programa.

Además, simples adiciones de otro trabajo no basado en el Programa al mismo Programa (o con un trabajo basado en el Programa) en un volumen de almacenamiento o medio de distribución no incluye al otro trabajo bajo el alcance de esta licencia.

3. Usted puede copiar y distribuir el Programa (o un trabajo basado en el, ver sección 2) en código objeto o en forma ejecutable bajo los términos de las secciones 1 y 2 antes mencionados y que además puede hacer una de las siguientes opciones:
  - a. Acompañarlo con el total código fuente, que debe ser distribuido bajo los términos de las secciones 1 y 2 en un medio común para software; o,
  - b. Acompañarlo con una oferta escrita, válida como mínimo por 3 años, a un tercero, por un cargo no mayor al costo de la distribución física del código fuente, que será distribuida bajo los términos de las secciones 1 y 2 en un medio común para software; o,
  - c. Acompañarlo con la información que usted recibió tal como la oferta para distribuir el correspondiente código fuente. (Esta alternativa se permite solamente para distribución no comercial y solamente si usted recibió el programa en código objeto o en forma ejecutable con tal oferta, de acuerdo a la subsección b antes mencionada).

El código fuente para un trabajo significa la mejor forma para hacerle modificaciones. Para un trabajo ejecutable, el código fuente completo implica todo el código fuente para todos los módulos que contiene, además de cualesquiera archivos de definición de interfaz asociados, mas los scripts utilizados para controlar la compilación e instalación del ejecutable. Sin embargo, como una excepción especial, el código fuente distribuido no necesita incluir nada que normalmente sea distribuido (en cualquier fuente o forma binaria) con los componentes (compiladores, kernel, etc.) del sistema operativo en el que corre el ejecutable, a menos que el mismo componente acompañe el ejecutable.

Si la copia y distribución de ejecutables o código objeto se hace desde un lugar designado, se asume este lugar como el punto de distribución del código fuente y las terceras partes no están obligados a copiar el fuente con el código objeto.

4. Usted no podrá copiar, modificar, sublicenciar, o distribuir el Programa excepto como expresamente se ha dicho bajo esta Licencia. Cualquier intento de copiar, modificar, sublicenciar o distribuir el Programa debe evitarse, y automáticamente se terminarán sus derechos de esta Licencia. Sin embargo, aquellas partes que recibieron copias, o derechos, de usted bajo esta Licencia no tendrán sus Licencias canceladas hasta tanto esas partes continúen bajo los términos de la Licencia.
5. Usted no está obligado aceptar esta licencia ya que no la ha firmado. Sin embargo ninguna otra cosa le permite modificar o distribuir el Programa o sus trabajos derivados. Tales acciones están prohibidas por la ley si usted no acepta esta Licencia. Por lo tanto al modificar o distribuir el Programa (o cualquier trabajo basado en el), usted indica su aceptación de esta Licencia, y todos sus términos y condiciones para copiar, modificar y distribuir el Programa o trabajos basados en el.
6. Cada vez que usted redistribuya el Programa (o cualquier trabajo basado en el) el beneficiario automáticamente recibe una Licencia del Licenciario original para copiar, distribuir o modificar el Programa sujeto a estos términos y condiciones. Usted no debe imponer ninguna restricción adicional al ejercicio de los derechos que se otorga al beneficiario. Usted no es responsable de que terceras partes cumplan con esta licencia.
7. Si, como consecuencia de un juicio de corte o alegación de infringir la patente o por cualquier otra razón (no limitada a la patente), se imponen sus condiciones (sea por orden de la corte, un acuerdo o cualquier otra) que contradice las condiciones de esta licencia, eso no lo excusa a usted de estas mismas condiciones. Si usted no puede distribuir de tal manera que satisfaga simultáneamente sus obligaciones bajo esta licencia y cualquier otras obligaciones pertinentes, entonces usted no podrá distribuir el Programa. Por ejemplo, si una licencia de patente no permitiese una redistribución libre de los derechos de autor del Programa por todos aquellos que reciben copias directa o indirectamente a través de usted, entonces la única manera de que usted satisfaga ambas sería abstenerse completamente de distribuir el Programa.

Si cualquier parte de esta sección se considera inválida o no aplicable bajo cualquier circunstancia, el balance de la sección se intenta aplicar y la sección como un todo se aplica en otras circunstancias.

No es el propósito de esta sección inducirlo a usted a infringir cualquier patente u otros reclamos sobre los derechos de propiedad o disputar la validez de tales reclamos; esta sección tiene el único propósito de proteger la propiedad del sistema de distribución de software gratis, que se implementa por prácticas de

licencias públicas. Mucha gente ha hecho generosas contribuciones al amplio rango de software distribuido a través del sistema confiando en la consistencia de la aplicación; depende del autor/donante decidir si él es capaz de distribuir el software a través de cualquier otro sistema y un licenciataria no puede imponer esa alternativa.

Esta sección intenta aclarar completamente lo que se considera es una consecuencia del resto de esta Licencia.

8. Si la distribución y/o uso del Programa se restringe en ciertos países ya sea por patentes o por interfaces copyrights, el dueño original del copyright quien coloca el Programa bajo esta Licencia puede agregar una limitación de distribución geográfica explícita excluyendo esos países, de manera que la distribución sea permitida solamente en o entre países no excluidos. En tal caso, la Licencia incorpora la limitación como si estuviese escrita en su cuerpo.
9. La Fundación del Software gratis puede publicar versiones nuevas y/o revisadas de la Licencia Pública General de vez en cuando. Tales nuevas versiones serán similares en espíritu a la presente, pero pueden diferir en detalle al enfocar nuevos problemas.

A cada versión se le da un número distintivo. Si el Programa especifica un número de versión de esta Licencia que se aplica a esa y a cualquier otra versión posterior, usted tiene la opción de seguir los términos y condiciones de esa versión o de cualquier otra posterior publicada por Fundación del Software gratis. Si el programa no especifica un número de versión de esta Licencia, usted puede elegir cualquier versión publicada por la Fundación del Software gratis.

10. Si usted desea incorporar partes del Programa en otros programas gratis cuyas condiciones de distribución sean diferentes, escríbale al autor para solicitar permiso. Para software cuyo copyright sea de la Fundación del Software gratis, escríbale a la Fundación; algunas veces hacemos excepciones para esto. Nuestra decisión estará guiada por las dos metas de preservar la condición gratis de todos los derivados de nuestro software gratis y de promocionar el software compartido y el reuso del mismo.

#### SIN GARANTIA

11. YA QUE EL PROGRAMA ES LICENCIADO LIBRE DE CARGO, NO HAY GARANTIA PARA EL PROGRAMA, HASTA LO QUE PERMITE LA LEY. EXCEPTO CUANDO SE ESTABLEZCA LO CONTRARIO POR LOS DUEÑOS DEL COPYRIGHT Y/O TERCERAS PARTES PROVEAN EL PROGRAMA

“COMO ES” SIN GARANTIA DE NINGUN TIPO, SEA EXPRESA O IMPLICITA, INCLUIDA, PERO NO LIMITADA A LAS GARANTIAS IMPLICITAS DEL ASPECTO MERCANTIL Y QUE SE AJUSTE A UN PROPOSITO PARTICULAR. EL RIESGO COMPLETO DE LA CALIDAD Y DESEMPEÑO DEL PROGRAMA ES SUYO. SI EL PROGRAMA SE COMPRUEBA QUE ESTA DEFECTUOSO, USTED ASUME EL COSTO DE TODOS LOS SERVICIOS NECESARIOS, REPARACIONES O CORRECCIONES.

12. EN NINGUN CASO AL MENOS QUE LA LEY LO EXIJA O EN ACUERDO AL ESCRITO NINGUN DUEÑO DEL COPYRIGHT O CUALQUIER OTRA PARTE QUIEN PUEDA MODIFICAR Y/O REDISTRIBUIR EL PROGRAMA TAL COMO SE PERMITE EN ESTA LICENCIA, LO RESPONSABILIZARA A USTED POR DAÑOS, INCLUYENDO CUALQUIER DAÑO GENERAL, ESPECIAL, INCIDENTAL O A CONSECUENCIA DEL USO O INCAPACIDAD DE UTILIZAR EL PROGRAMA (INCLUYENDO PERO NO LIMITADO A LA PERDIDA DE DATOS O DE DATOS SUMINISTRADOS INCORRECTAMENTE O PERDIDAS OCASIONADAS POR USTED O POR TERCERAS PARTES O UNA FALLA DEL PROGRAMA PARA OPERAR CON CUALESQUIERA OTROS PROGRAMAS), AUN CUADO EL DUEÑO U OTRA PARTE HAYA SIDO NOTIFICADO DE LA POSIBILIDAD DE TALES DAÑOS.

FIN DE LOS TERMINOS Y CONDICIONES

### **A.1.3 Anexo: Cómo aplicar estos términos a sus nuevos programas.**

Si usted desarrolla un nuevo programa, y usted quiere que sea aprovechado al máximo por el público, la mejor manera de lograrlo es convirtiéndolo en software gratis que cualquiera pueda redistribuir y cambiar bajo estos términos.

Para hacerlo, anexe las siguientes notas al programa. Es más seguro anexarlo al comienzo de cada archivo fuente para que más efectivamente participe la exclusión de la garantía; y cada archivo debe llevar por lo menos la línea del “copyright “ y un apuntador a donde se consigue la nota completa.

< Una línea para dar el nombre del programa y una breve idea de lo que hace.>

Copyright © 19yy <nombre del autor>

Este programa es un software gratis; usted puede redistribuirlo y/o modificarlo bajo los términos de la Licencia Pública General como lo a

publicado la Fundación del Software Gratis; sea la versión 2 de la Licencia, o (a su elección) cualquier versión posterior.

Este programa se distribuye con la esperanza que sea útil, pero SIN NINGUNA GARANTIA; sin ni siquiera la garantía implícita del ASPECTO MERCANTIL o AJUSTE A PROPOSITOS PARTICULARES. Véase la Licencia Pública General para mas detalles.

Usted debe haber recibido una copia de la Licencia Pública General junto con este programa; si no, escriba a la Fundación del Software Gratis, Inc, 675 Mass Ave, Cambridge, MA 02139, USA.

Además agregue información sobre como contactarlo a usted por correo electrónico y postal.

Si el programa es interactivo, imprima una corta nota como esta cuando comience en el modo interactivo:

Gnomovision version 69, Copyright © 19yy nombre del autor Gnomovision viene SIN GARANTIA ABSOLUTA; para detalles escriba 'show w'.

Este es un software gratis, y usted está invitado a redistribuirlo bajo ciertas condiciones; escriba 'show c' para detalles.

Los comandos hipotéticos 'show w' y 'show c' deberían mostrar las partes correspondientes de la Licencia Publica General. Naturalmente, los comandos que usted utilice puede ser llamados de otra forma que no sea 'show w' y 'show c'; podrían ser incluso llamados por el mouse o ítems del menú, lo que se adapte a su programa.

Usted además podría hacer que su compañía (si usted trabaja como programador) o su escuela, firme una "renuncia del copyright" del programa, si es necesario. Aquí hay un ejemplo; después de los nombres:

Yoyodine, Inc., renuncia a todos los intereses del copyright del programa 'Gnomovision' (que incluye al compilador) escrito por James Hacker.

<firma de Ty Coon>, 1 de Abril de 1989  
Ty Coon, Presidente de Vice

Esta Licencia Publica General no permite incorporar su programa dentro de programas propietarios. Si su programa es una biblioteca de subrutinas, usted puede considerarlo más útil el permitir enlaces de aplicaciones propietarias con la

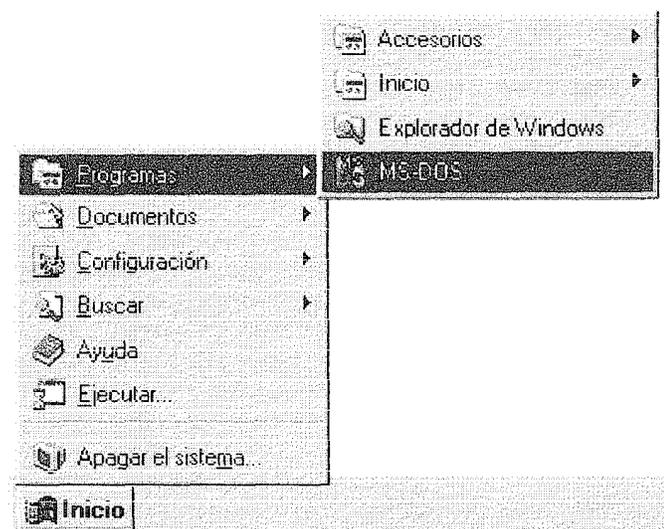
biblioteca. Si esto es lo que usted quiere hacer use la Licencia Publica General de Biblioteca GNU en vez de esta Licencia.

## A.2 Instalando el sistema CAD ALLIANCE bajo el sistema operativo WINDOWS 95.

La instalación del sistema portado CAD ALLIANCE bajo es sistema operativo WINDOWS 95 es muy sencillo.

Para instalar el sistema portado ALLIANCE en su disco duro, siga los siguientes pasos:

1. Inserte el CD-ROM del "ALLIANCE bajo WINDOWS 95" en su unidad de CD.
2. Abra una ventana MS-DOS, haciendo "click" en el ítem "MS-DOS" perteneciente al menú de "Programas" del menú de "Inicio" de la barra de herramientas de WINDOWS 95.



3. Cámbiese a la unidad donde se encuentra el CD-ROM del "ALLIANCE bajo WINDOWS 95".

Por ejemplo:

**D:**

4. Ejecute el programa para la instalación del ALLIANCE (install-alliance.bat), indicando la unidad donde se encuentra el CD-ROM del "ALLIANCE bajo WINDOWS 95".

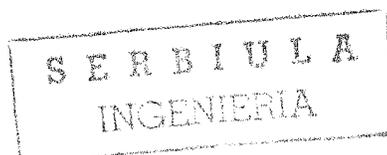
**D:>install-alliance d:**

5. Cuando finalice la instalación, usted debe definir y actualizar un conjunto de variables de entornos en el archivo AUTOEXEC.BAT, necesarias para la ejecución del sistema CAD ALLIANCE.

```
SET TOP=/alliance
SET MACHINE=win95
PATH=c:\alliance;c:\alliance\archi\win95\bin;c:\bin;c:\usr\bin;c:\usr\local\bin;%PATH%
SET MANPATH=/alliance/man:/usr/man:/usr/local/man
SET MBK_CATA_LIB=/alliance/celis/scr:/alliance/celis/ring
SET MBK_WORK_LIB=.
SET MBK_IN_LO=vst
SET MBK_OUT_LO=vst
SET MBK_IN_PH=ap
SET MBK_OUT_PH=ap
SET VH_MAXERR=10
SET VH_PATSFY=pat
```

6. Reinicie el equipo.

De esta manera queda instalado, no solo la herramienta ALLIANCE, sino también las aplicaciones necesarias para la emulación del entorno UNIX y el compilador ANSI C y ANSI C++.



## A.3 Ejecutando el tutorial AMD2901 con el ALLIANCE bajo Windows 95.

### A.3.1 Uso de la herramienta ASIMUT para la validación de la sintaxis del modelo de comportamiento.

```
C:\ALLIANCE\ARCHI\WIN95\BIN\ASIMUT.EXE -b -c and
5 x 12
C:\Alliance\tutorials\AMD2901>asimut -b -c and

      0      0000 0      0      0000000000
      0      0      00  000      0      00  0
      000      00  0      0      000 00 000      000 0000      00
      0  00      0000      0000      000 00 00      00      00
      0  00      0000      00      00  00  00      00      00
      0  00      000  00      00  00  00      00      00
      0000000 0      00  00      00  00  00      00      00
      0  00  00      00  00      00  00  00      00      00
      0  00  000  0      00  00  00  00      00      00
      0000      0000 0  0000      000000 000 000      0000 00      000000

A SIMULATION Tool

Alliance CAD System 3.0,          asimut v1.93
Copyright (c) 1991..1994, MASI, CAD-VLSI Team
E-mail support:                   cae-vlsi@masi.ibp.fr

Paris, FRANCE, Europe, Earth, Solar system, Milky Way, ...
Initializing ...
Searching 'and' ...
GEN : Compiling 'and.vbe' (Behaviour) ...
Making GEX ...

C:\Alliance\tutorials\AMD2901>
```

### A.3.2 Uso de la herramienta ASIMUT para la simulación de la descripción lógica del circuito.

```

C:\ALLIANCE\ARCHI\WIN95\BIN\ASIMUT EXE -b amd
5 x 12
C:\Alliance\tutorials\AMD2901>asimut -b amd pattern result_hoh

      0      0000 0      0      000000000
      0      0      00  000
      000  00  0      0
      000  000      000 00 000  000 0000
      0 00  0000  0000  000 00 00  00 00  00
      0 00  0000  00  00 00 00  00 00  00
      0 00  000  00  00 00 00  00 00  00
      0000000 0      00 00  00 00 00  00 00  00
      0      00 00  00 00  00 00 00  00 00  00
      0      00 000  0  000  00 00 00  00 000
      0000  0000 0 0000 00000 0000 000  0000 00  000000

          A SIMULATION TOOL

Alliance CAD System 3.0,      asimut v1.93
Copyright (c) 1991..1994, MASI, CAD-VLSI Team
E-mail support:      cao-vlsi@masi.ibp.fr

Paris, FRANCE, Europe, Earth, Solar system, Milky Way, ...
Initializing ...
Searching 'and' ...
GEN : Compiling 'and.vbe' (Behaviour) ...
Making GEN ...
Searching pattern file : 'pattern' ...
Restoring ...
Linking ...
###---- processing pattern 0 ----###
###---- processing pattern 1 ----###
###---- processing pattern 2 ----###
###---- processing pattern 3 ----###
.....
###---- processing pattern 496 ----###
###---- processing pattern 499 ----###
###---- processing pattern 500 ----###
C:\Alliance\tutorials\AMD2901>

```



### A.3.4 Uso de la herramienta ASIMUT para la simulación del modelo estructural.

```

C:\VALLIANCE\ARCHI\WIN95\BIN\ASIMUT EXE chip.p
5 x 12
C:\Valliance\tutorials\AND9901\asimut_chip_pattern_result.txt

      0      0000 0      0      000000000
      0      0      00      000      000 00 00      00      00      00
      000      00      0      0      000 00 000      000      00000
      000      000      00000 0000      000 00 000      000      0000
      0 00      00000      0000      000 00 00      00      00      00
      0 00      0000      00      00 00 00      00      00      00
      0 00      000      00      00 00 00      00      00      00
      0000000 0      00      00      00 00 00      00      00      00
      0      00 00      00      00      00 00 00      00      00      00
      0      00 000      0      00      00 00 00      00      000
      0000      0000 0      0000      00000 0000 000      0000      000000

      R SIMULATION TOOL

      Valliance CAD System 3.0,          version v1.03
      Copyright (c) 1991...1994, HESI, CSI-MAT Team
      E-mail support:                   cao-valliance@ihp.fr

      Paris, FRANCE, Europe, Earth, Solar system, Milky Way, ...

      Initializing ...
      Searching 'chip' ...
      Compiling 'chip' (Structural) ...
      Warning 2 : consistency checks will be disabled
      Flattening the root figure ...
      Searching 'na3_y' ...
      BEH : Compiling 'na3_y.vbe' (Behaviour) ...
      Making BEH ...
      ...
      Searching 'pot_sp' ...
      BEH : Compiling 'pot_sp.vbe' (Behaviour) ...
      Making BEH ...
      Searching pattern file : 'pattern' ...
      Restoring ...
      Linking ...
      ###----- processing pattern 0 -----###
      ...
      ###----- processing pattern 500 -----###
      C:\Valliance\tutorials\AND9901>
  
```





### A.3.7 Uso de la herramienta LYNX para la extracción del netlist de compuertas del *heart* del *chip*.

```

C:\VALLIANCE\ARCHI\WIN95\BIN\LYNX EXE heart
5 x 12
C:\VALLIANCE\Archivos de trabajo\WIN95\BIN>lynx heart

000000
00
00
00      00000 000 000 000      0000 000
00      00  0  000  0  00  0
00      00  0  00  00  00  0
00      00  0  00  00  000
00      0  000  00  00  0  00
00      0  00  00  00  0  00
000000000 00  0  0000 0000 000 0000
00  0
000

Netlist extractor

Alliance CAD System 3.0.      lynx 1.10
Copyright (c) 1994, MCSI, CAC-VLSI Team
E-mail support: cac-vlsi@rsi.ihp.fr

--> Extracts figure heart
<-- done !
C:\VALLIANCE\Archivos de trabajo\WIN95\BIN>

```



### A.3.9 Uso de la herramienta RING para el ruteado del chip.

```

C:\ALLIANCE\ARCHI\WIN95\BIN\RING EXE chip chip
5 x 12
C:\Alliance\tutorials\WIN95\BIN>ring chip chip

0000000 0 0000 0
00 00 000 00 00
00 00 0 00 0
00 00 000 000 00
00 00 0000 000 0 00
00000 00 00 00 00000
00 00 00 00 00 00 0 00
00 00 00 00 00 00 0 00
00 00 00 00 00 00
00 00 00 00 00 00
00000 000 00000 0000 0000 0000

PAD ring router

Alliance CAD System S.A. ring 2.9
Copyright (c) 91-92, MAGI, CAD-MAGI Team
E-mail support: cas-vlsi@magi.lbp.fr

o reading netlists, layout views of core and pads.
warning 2 : consistency checks will be disabled
o reading file of parameters, including the placements of pads.
o making equipotential list.
o making the first placement of pads.
o filling data internal structures.
o reading the connectors positions of the core.
o computing the best placement of the pads.
o reading the connectors positions of the pads.
o routing deformation of connectors.
o routing supply tracks.
o routing equipotentials.
o compressing channels.
o saving in MCK data structure.

lucky, no error.
C:\Alliance\tutorials\WIN95\BIN>

```



### A.3.11 Uso de la herramienta LYNX para la extracción del netlist de compuertas del *chip*.

```

C:\Alliance\ARCHI\WIN95\BIN\LYNX.EXE chip
C:\Alliance\tutorials\NH02901>lynx chip

      000000
      00
      00
      00      00000 000 000 000      0000 000
      00      00 0 000 0 00 0
      00      00 0 00 00 00 0
      00      00 0 00 00 000
      00      0 000 00 00 0 00
      00      0 00 00 00 0 00
      000000000 00 0 0000 0000 000 0000
      00 0
      000

Netlist extractor

Alliance CAD System 3.0,      lynx 1.10
Copyright (c) 1994, NH01, CAD-VLSI Team
E-mail support:  cad-vlsi@asi.hq.fv

---> Extracts figure chip
<--- done!
C:\Alliance\tutorials\NH02901>

```



### A.3.13 Uso de la herramienta ASIMUT para la simulación del chip.

```

C:\ALLIANCE\ARCHI\WIN95\BIN\ASIMUT_EXE chip.p
5 x 12
C:\Alliance\tutorials\AM02901>asimut chip pattern vbe_pattern

      0      0000 0      0      000000000
      0      0      00  000      0      00  0
      000  00  0      0      000 00 000  000 0000  00
      000  000      000  000 00 000  000 00  00
      0 00  0000  0000  000 00 00  00  00  00
      0 00  0000  00  00 00 00  00  00  00
      0 00  000  00 00 00 00  00  00  00
      0000000 0      00  00  00 00 00  00  00
      0 00 00  00  00 00 00  00  00  00
      0 00 000  0 00 00 00  00  000  00
      0000  0000 0 0000  000000 0000 000 000  0000 00  000000

          A SIMULATION Tool

          Alliance CAD System 3.0,          asimut v1.93
          Copyright (c) 1991...1994, MMSI, CAD-MLSI Team
          E-mail support:                   cae-vis@ncei.ibp.fr
          Paris, FRANCE, Europe, Earth, Solar system, Milky Way, ...

Initializing ...
Searching 'chip' ...
Compiling 'chip' (Structural) ...
Flattening the root figure ...
Searching 'a4.y' ...
GEN : Compiling 'a4.y.vbe' (Behaviour) ...
taking GEN ...
...
Searching 'pi.sp' ...
GEN : Compiling 'pi.sp.vbe' (Behaviour) ...
taking GEN ...
Searching pattern file : 'pattern' ...
factoring ...
linking ...
###----- processing pattern 0 -----###
...
###----- processing pattern 500 -----###
C:\Alliance\tutorials\AM02901>

```

### A.3.14 Uso de la herramienta DESB para hacer la abstracción funcional del netlist de transistores del chip.

```

C:\VALLIANCE\ARCHI\WIN95\BIN\DESB.EXE chip -i -v
5 x 12
C:\VALLIANCE\TUTORIALS\95\DESB\DESB.CHIP -I -V
      000          000
      00          00
      00          00
      00 00      00000 000000 00 00
      00 000 0 0 00 0 000 00
      00 00 00 00 000 00 00
      00 00 00000000 0000 00 00
      00 00 00 0000 0000 00 00
      00 00 00 0 0 000 00 00
      00 000 00 00 00 00 000 00
      00 000 0000 0 0000 000 00
      Functional Abtractor
      Alliance CAD System S.E.      desb 2.06
      Copyright (c) 91-94, NRSI, CRC-VLSI Team
      E-mail support:   car-vlsi@resist.lbp.fr
(NES) Reading file 'chip.in'
(NES) Loading the figure          00:00s  ut:6249407-26.-0  st:07'20
.2
(NES) Flattening the figure      00:00s  ut:3206487-58.-5  st:25'22
W-1.-2
(NES) Transistor netlist checking 00:00s  ut:2380687-19.-1  st:11'24
1'37.2
(NES) Making gates              00:00s  ut:9941475.7  st:02'24.
0
(NES) Gones/Paths 1200 / 5691    00:00s  ut:300934749.0  st:02'07.
-21.-6
(NES) External connectors orientation 00:00s  ut:2953997-17.-9  st:11'45
02'-05.-3
(NES) Dual gates detection      00:00s  ut:534486752.5  st:23'58.
-4.-8
(NES) Latches detection        00:00s  ut:565050716.6  st:43'04.
-24.-6
(NES) Functional analysis       00:00s  ut:314506721.2  st:46'48.
-33.0
(NES) Gates characterisation     00:00s  ut:445601748.0  st:50'22.
-41.-2
(NES) Master Slave detection    00:00s  ut:319413725.-5  st:36'32
10 -42.-7
(NES) TOTAL DISASSEMBLING TIME  00:00s  ut:42115739.0  st:047'00.
53.-1
-----
(NES) Erasing the transistor netlist 00:00s  ut:78828705.6  st:53'05'06
.2
(NES) Generating the Wvdl Data File 00:00s  ut:422757756.3  st:347'08.
-53.-1
(NES) Execution COMPLETED
-----
(WNR) 41 6 transistors are used as a resistance
(WNR) 61 216 transistors are always off
(WNR) 121 4 conflicts may occur in the circuit
(WNR) 131 8 signals may be HZ
See file 'chip.debr' for more information
-----
C:\VALLIANCE\TUTORIALS\95\DESB>

```

### A.3.15 Uso de la herramienta PROOF para chequear la equivalencia formal entre la descripción del comportamiento del *chip* abstraído y la inicial.

```

C:\ALLIANCE\ARCHI\WIN95\BIN\PROOF.EXE -d amd
5 x 12
C:\Alliance\tutorials\AMD2901>proof -d amd chip

      00000000                                000
      00 00                                00 00
      00 00                                00 00
      00 00 00 00 00 00 00 00 00 00 00 00
      00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
      000000 00 00 00 00 00 00 00 00 00 00 00
      00 00 00 00 00 00 00 00 00 00 00 00
      00 00 00 00 00 00 00 00 00 00 00 00
      00 00 00 00 00 00 00 00 00 00 00 00
      00 00 00 00 00 00 00 00 00 00 00 00
      000000 0000 000 000 000000

          Formal Proof
Alliance CAD System 2.3,      proof 2.02
Copyright (c) 90-94, MEGI, CAD-VLSI Team
E-mail support:   cae-vlsi@nsl.lhp.fr
----- Environment -----
MAC_HUNK_LIB          = .
MAC_CWTR_LIB          = /alliance/calls/src/alliance/cwtr/ring
----- Files, Options and Parameters -----
-list VHDL File       = amd.vbe
-record VHDL File     = chip.vbe
The auxiliary signals are erased
Errors are displayed
-----
Compiling 'amd' ...
Compiling 'chip' ...
Running abi ordonnanceur on 'amd' .....
Running Abi2Bdd on 'amd' .....
Running Abi2Bdd on 'chip' .....
-----
          Formal proof with ordered Binary Decision Diagrams between
          'amd' and 'chip'
-----
PRIMARY OUTPUT
AUXILIARY SIGNAL
REGISTER SIGNAL
EXTERNAL BUS
INTERNAL BUS
-----
          Formal Proof : OK
C:\Alliance\tutorials\AMD2901>

```

### A.3.16 Uso de la herramienta S2R para la finalización del chip.

```

C:\VALLIANCE\ARCHI\WIN95\BIN\S2R.EXE -c chip am...
5 x 12
C:\Valliance\tutorials\AM2901\s2r -c chip am2901

          0000
            0 00
             00 00
          00000 000 00 000 000
         00 0 0 00 000 00
        000 0 00 00
         0000 0 00
          0000 0 00
         0 000 0 0 00
        00 00 00000 00
       0 0000 00000 0000

Symbolic to Real Layout converter

Alliance CAD System 3.0.      s2r 3.4
Copyright (c) 91-99, ANCI, CAD-VLSI Team
E-mail support:   cad-vis@ncsi.ibp.fr

o loading technology file : /alliance/etc/pral00.rds
o loading all levels of symbolic layout : chip
o removing symbolic data structure
o layout post-treating without connector, with switch.
o replacing black boxes
o saving am2901.rtl

C:\Valliance\tutorials\AM2901>

```

# Bibliografía

**[ALLIANCE]** ALLIANCE : una herramienta CAD para el diseño VLSI.  
<http://cao-vlsi.ibp.fr/index.html>

**[ALLIANCE-1]** La Herramienta CAD ALLIANCE.  
<http://nswt.tuwien.ac.at/htdocs/internet/unix/cad/ALLIANCE.html>

**[ALLIANCE-2]** La Herramienta CAD ALLIANCE.  
<http://www-asim.lip6.fr/alliance/index.html>

**[ALLIANCE-DOC]** Documentación contenida en el paquete de la Herramienta CAD ALLIANCE y los manuales en línea de la herramienta ALLIANCE.

**[OSW97]** Documentación facilitada por el prof. José Oswaldo Cadenas.

**[CYGNUS]** CYGNUS Solutions. Compañía proveedora de software y herramientas de desarrollo multiplataforma.  
<http://www.cygnus.com/>