

Modelado No Lineal de Dispositivos Activos Microondas Utilizando Verilog-A

Andrea Gutiérrez Ontañón⁽¹⁾, Tomás Fernández Ibáñez⁽¹⁾, Angel Mediavilla Sánchez⁽¹⁾, Carlos Camacho-Peñalosa⁽²⁾, Enrique Márquez-Segura⁽²⁾, Virginia Comino García⁽²⁾

E-mail Primer Autor, E-mail Segundo Autor, E-mail Tercer Autor, etc.

⁽¹⁾ Dpto. de Ingeniería de Comunicaciones. Universidad de Cantabria. E.T.S.I.I.T, Avda Los Castros s/n, 39005, Santander

⁽²⁾ Dpto. de Ingeniería de Comunicaciones. Universidad de Málaga. E.T.S.I. de Telecomunicación Campus de Teatinos 29071 Málaga

Abstract- Nowadays, as technology evolves, obtaining accurate nonlinear models is of prime importance for designers of Power Amplifiers. This document contains information concerning Verilog-A model capabilities and efficiency with non-linear electron devices. The Verilog-A abilities have been implemented on Agilent-ADS CAD simulator. A model defined with Verilog-A can be simulated in different software platforms regardless the particular simulator used to create the device models. This is the main advantage of Verilog-A, it can be used in different simulation platforms and is not a proprietary patent.

I. INTRODUCCIÓN

La calidad de los modelos pone de manifiesto la precisión de un simulador. Aunque los simuladores actuales poseen una gran variedad de modelos, la tecnología de dispositivos evoluciona continuamente, requiriendo nuevos dispositivos. La creación de modelos no lineales es costosa, por esta razón se están empezando a utilizar nuevas técnicas para modelado no lineal. Este es el caso del lenguaje de descripción hardware Verilog-A, el cual permite describir componentes y sistemas analógicos de manera sencilla y con la ventaja de que el usuario puede modificar fácilmente las ecuaciones del modelo, y dicho modelo puede exportarse a cualquier plataforma de simulación que soporte el lenguaje.

Con este trabajo se pretende llevar a cabo un estudio profundo del lenguaje Verilog-A en el modelado Gran Señal de dispositivos semiconductores de Microondas. Para ello, se utilizará el simulador ADS, desarrollando los códigos Verilog-A correspondientes. Aunque los resultados obtenidos serán fácilmente trasladables a otros simuladores comerciales que soporten Verilog-A como lenguaje de programación.

II. VERILOG-A

Verilog-A es un lenguaje de descripción hardware, analógico, de alto nivel y estandarizado por el IEEE. Pretende poder ser aplicado a la descripción tanto de sistemas eléctricos como de no eléctricos.

Un modelo creado con Verilog-A presentará la misma apariencia y funcionalidad que un modelo ya definido internamente en el simulador pero con la ventaja de que el usuario final puede modificar las ecuaciones del modelo cuando desee.

A. Arquitectura Verilog-A

Los componentes clave de la arquitectura son un compilador que genera dinámicamente una librería enlazable y un entorno de tiempo de ejecución RTE (*run-time environment*) que es distinto para cada simulador particular. El creador del modelo proporciona la relación constitutiva entre las entradas y salidas, los nombres de los parámetros y rangos, mientras que el compilador de Verilog-A maneja las interacciones necesarias entre el modelo y el simulador.

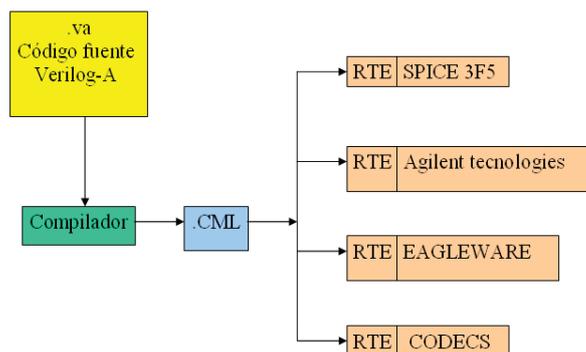


Fig. 1. Portabilidad de los modelos creados con Verilog-A

Como se ve en la Fig.1 el fichero con el código en Verilog-A tiene extensión .va, el compilador produce un archivo compilado (.CML) del modelo independiente, que presenta portabilidad, esto es, aunque haya sido desarrollado por un simulador concreto, puede compartirse por muchos tipos de simuladores obteniéndose los mismos resultados. Además puede someterse a cualquier tipo de análisis como continua, parámetros S o balance armónico.

B. Lenguaje Verilog-A

Verilog-A hace uso de los llamados *módulos* para hacer la descripción tanto estructural como del comportamiento de los sistemas analógicos y de sus componentes. El comportamiento de cada módulo puede describirse matemáticamente en términos de sus terminales y de los parámetros externos aplicados al módulo. Esas descripciones pueden usarse en muchas disciplinas como la eléctrica, mecánica, dinámica de fluidos y termodinámica.

En la Fig. 2 puede verse la estructura de un fichero tipo Verilog-A, el cual merece algunos comentarios:

- Los archivos de extensión *.vams* son insertados por el compilador en el código y éstos pueden contener constantes y funciones matemáticas que van a usarse en el fichero, o bien contenidos específicos que indican que el modelo es eléctrico (*disciplines.vams*).

- Dentro del módulo se van a incluir los puertos o nodos del modelo y todos los parámetros que intervienen en el mismo, bien parámetros de las ecuaciones de sus elementos no lineales o bien valores de componentes lineales.

- En el módulo también es necesario definir el bloque analógico, en el que se incluyen: las posibles variables locales que pueden utilizarse dentro del mismo, las ecuaciones de los elementos y todas las relaciones entre tensiones y corrientes del circuito. El operador de asignación es <+ como se verá en los modelos creados.

```
// Inserción de los archivos .vams
//Comienza la declaración de la interfaz del módulo
//Nombre del módulo (module nombre(puerto1,...))
//Definición de puertos
//Parámetros
//Descripción del comportamiento del modelo
//Variables locales
//Comienza el Bloque Analógico (analog begin)
//Se definen las ecuaciones del modelo
//Se describen las relaciones existentes entre las tensiones
//e intensidades del circuito mediante las asignaciones correspondientes
//Se cierra el Bloque Analógico (end)
//Se cierra el archivo, y por tanto el módulo (endmodule)
```

Fig. 2. Estructura de un fichero “tipo” Verilog-A

III. ALGUNOS MODELOS CREADOS CON VERILOG-A

A. Diodo de unión p-n

El circuito equivalente de un diodo de unión p-n se muestra en la Fig. 3, donde se han marcado los nodos que lo delimitan *ánodo* y *cátodo*, además se necesita un nodo *interno*.

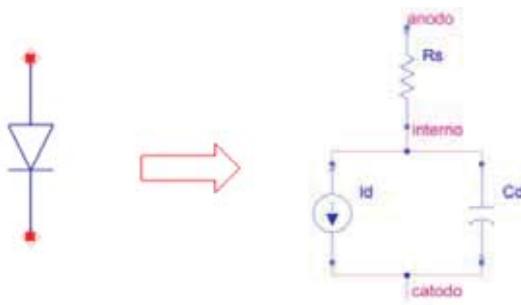


Fig. 3. Circuito equivalente de un diodo de unión p-n

Aparecen dos elementos no lineales en paralelo, la ecuación que describe la fuente no lineal *Id* viene dada por (1).

$$I_d = I_{ss} \cdot e^{(\alpha \cdot V_d - 1)} \quad (1)$$

Las ecuaciones que describen la capacidad vienen dadas por (2).

$$C_r(v) = C_o \cdot \sqrt{\frac{V_o}{V_o - v}} \quad v < \beta V_o \quad (2)$$

$$C_f(v) = C_r(\beta V_o) + C_r(\beta V_o) \cdot (v - \beta V_o) \quad v \geq \beta V_o$$

Para implementar la capacidad no lineal, se va a definir la corriente que las atraviesa en función de su carga no lineal, como se indica en (3).

$$i = \frac{d}{dt} Q(v) \quad (3)$$

Por tanto, las ecuaciones de la carga no lineal son las que se indican en (4), donde se ha integrado la capacidad.

$$Q_r(v) = -2C_o \cdot \sqrt{V_o \cdot (V_o - v)} \quad v < \beta V_o$$

$$Q_f(v) = \frac{C_o}{\sqrt{1 - \beta}} (v - \beta V_o + \frac{(v - \beta V_o)^2}{4V_o(1 - \beta)}) + Q_r(\beta V_o) \quad v \geq \beta V_o \quad (4)$$

Puesto que el diodo está limitado por los nodos *ánodo* y *cátodo*, estos serán los puertos que delimitan el dispositivo, y tendremos entonces que:

```
module diodo(anodo, catodo);
```

La tensión *Vd* entre los extremos de la fuente de intensidad, vendrá dada por:

```
Vd = V(interno, catodo);
```

La ecuación (1) de la corriente *Id* se transcribe en Verilog-A como:

```
Id = Is * (limexp(Vd * alpha) - 1);
```

donde *limexp* se utiliza para mejorar la convergencia de la función exponencial.

Las asignaciones serán por tanto las siguientes:

```
I(interno, catodo) <+ Id + ddt(Qd);
V(anodo, interno) <+ I(anodo, interno) * Rs ;
```

Con los resultados de las simulaciones veremos que el modelo de diodo creado funciona correctamente.

ANÁLISIS DE CONTINUA

Con este análisis se obtiene la curva característica I/V del diodo, de esta forma se probará el funcionamiento tanto de los algoritmos de convergencia para el cálculo de tensiones internas como la capacidad del simulador de analizar circuitos con fuertes no linealidades, como es el caso de la ecuación (1). Los resultados se muestran en la figura 4.

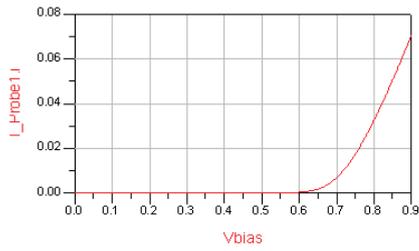


Fig. 4. Curva I/V del modelo de diodo creado en Verilog-A

ANÁLISIS DE PARÁMETROS S

Como resultado de este análisis se comprueba el comportamiento del modelo en pequeña señal, en la Fig. 5 puede observarse el parámetro S11 obtenido.

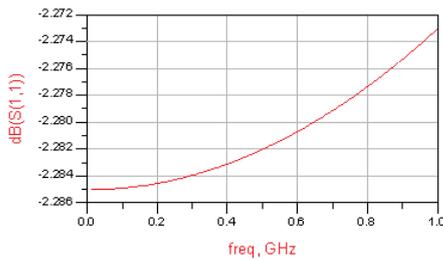


Fig. 5. Parámetro S11 del modelo de diodo creado en Verilog-A

B. Transistor HEMT de GaAs con Red Térmica

En este caso, el modelo a implementar es más complejo, se trata de un transistor HEMT cuyo circuito equivalente se muestra en la Fig. 6. La particularidad de este transistor es que se ha incluido en su modelo una red térmica para modelar el efecto de auto-calentamiento. Este efecto es importante tenerlo en cuenta ya que la potencia disipada, hace que aumente la temperatura interna del transistor, modifica las características dinámicas del dispositivo en operación normal.

La no linealidad más importante de este tipo de transistores es la fuente de corriente de drenador a surtidor I_{ds} , que viene definida por las ecuaciones en (5). Los otros elementos no lineales que aparecen en el circuito son la fuente I_{gs} , definida por (6), y las capacidades C_{gs} y C_{gd} cuyas cargas se definen en (7) y (8) respectivamente.

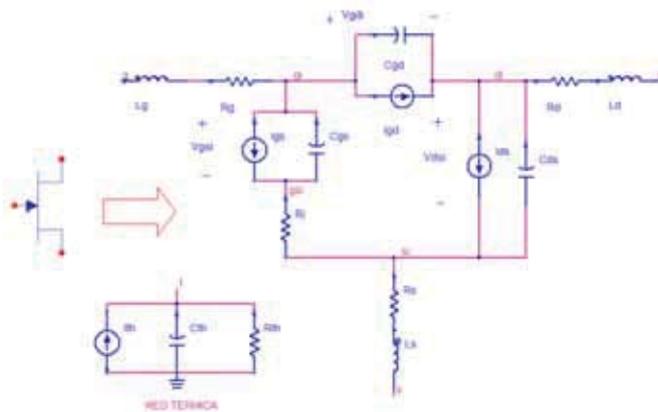


Fig. 6. Circuito equivalente del transistor HEMT

$$\begin{aligned}
 V_{gfi} &= V_{gsi} - V_{pf} \\
 V_{gitl} &= V_{gsi} - (V_p * (1 + V_t * 0.01) + \text{Gamma} * V_{dsi}) \\
 V_{gitlch} &= \ln(2 * \cosh(t2 * V_{gitl})) \\
 V_{gijlch} &= \ln(2 * \cosh(V_{gfi})) \\
 V_{giteff} &= \frac{1}{2 * n1} * (n2 * V_{gitl} + V_{gijlch}) \\
 M &= V_{gijlch}^{beta} \\
 I_{ds1} &= I_{dss} * (1 + \text{Temperatur} * 0.0025) * (V_{giteff})^{E + K * V_{gsi}} \\
 I_{ds2} &= (1 + S_s * \frac{V_{dsi}}{I_{dss}})^{M} * e^{\frac{M}{I_{dss}}} \\
 I_{ds} &= I_{ds1} * I_{ds2} * \tanh(SI * \frac{V_{dsi}}{I_{dss} * (1 - K_g * V_{gsi})})
 \end{aligned}
 \tag{5}$$

$$\begin{aligned}
 a_{s_igs} &= \text{Alpha_igs} * V_{gsi} \\
 b_{s_igs} &= \text{Alpha_igs} * V_{gsi} - M_{s_igs} \\
 I_{gs} &= I_{ns_igs} * (e^{M_{s_igs}} * (1 + b_{s_igs} + \frac{b_{s_igs}^2}{2}) - 1) & b_{s_igs} > 0 \\
 I_{gs} &= I_{ns_igs} * (e^{a_{s_igs}} - 1) & b_{s_igs} < 0 \\
 I_{gs} &= -I_{ns_igs} & a_{s_igs} < -20 \\
 & & b_{s_igs} < 0 \\
 & & a_{s_igs} \geq -20
 \end{aligned}
 \tag{6}$$

$$\begin{aligned}
 Q_{gs} &= \frac{C_{gso_cgs} * V_{bi_cgs}}{1 - \text{Gamma_cgs}} * (1 - (1 - \frac{V_{gsi}}{V_{bi_cgs}})^{(1 - \text{Gamma_cgs})}) & V_{gsi} < K_cgs * V_{bi_cgs} \\
 & + C_{gse_cgs} * V_{gsi} & \text{resto} \\
 Q_{gs} &= C_{gso_cgs} * (1 - K_cgs)^{-\text{Gamma_cgs}} * (V_{gsi} - K_cgs * V_{bi_cgs}) & \text{resto} \\
 & + C_{gse_cgs} * V_{gsi}
 \end{aligned}
 \tag{7}$$

$$\begin{aligned}
 Q_{gd} &= \frac{C_{gdo_cgs} * V_{bi_cgs}}{1 - \text{Gamma_cgs}} * (1 - (1 - \frac{V_{gdi}}{V_{bi_cgs}})^{(1 - \text{Gamma_cgs})}) & V_{gdi} < K_cgs * V_{bi_cgs} \\
 & + C_{gde_cgs} * V_{gdi} & \text{resto} \\
 Q_{gd} &= C_{gdo_cgs} * (1 - K_cgs)^{-\text{Gamma_cgs}} * (V_{gdi} - K_cgs * V_{bi_cgs}) & \text{resto} \\
 & + C_{gde_cgs} * V_{gdi}
 \end{aligned}
 \tag{8}$$

Como puede verse en la Fig. 6 el modelo está limitado por los puertos d, g, s, luego tenemos:
 module hemt (d, g, s);

Las definiciones de las tensiones del circuito son:
 $V_{gsi} = V(gi, gsi);$
 $V_{gdi} = V(gi, di);$
 $V_{dsi} = V(di, si);$
 $V_t = V(t);$

Donde $V(t)$ es la tensión en la Red térmica, esto es del nodo t a tierra (nodo 0, que no hace falta especificar). Esta red térmica se puede modelar con un circuito eléctrico equivalente como se ve en la Fig. 6

La corriente I_{th} se modela como la potencia disipada en el dispositivo. Aunque podría hacerse sin fuente controlada, hemos considerado que se trata de una fuente de intensidad controlada por intensidad de ganancia 1. Luego $I_{th} = I_{ds} * V_{dsi}$ que es lo mismo que $I(out) = I(in) * V_{dsi}$.

Para crear esta fuente controlada previamente se tienen que definir las ramas:

```
branch (di, si) in;
branch (t) out;
```

La potencia en función de la rama de entrada es:

```
Pdiss=I(in)*Vdsi;
```

Y en las asignaciones tendremos:

```
I(out)<+-Pdiss;
I(in) <+ Ids;
```

Es importante tener en cuenta que el orden de las ecuaciones es importante e influye en los resultados.

El resto de asignaciones serán las definiciones de todas las tensiones y corrientes del circuito, se puede comprobar que estas vienen dadas por:

```
I(gi,gsi) <+ Igs;
I(gi,di) <+ ddt(Qgd);
I(gi,gsi) <+ ddt(Qgs);
I(di,si) <+ Cds* ddt(Vdsi);
V(gsi,si) <+ I(gsi,si) * Ri;

V(si,s) <+ I(si,s) * Rs;
V(gi,g) <+ I(gi,g) * Rg;
V(d,di) <+ I(d,di) * Rd;

I(t) <+ Cth * ddt(V(t));
I(t) <+ V(t) / Rth;
```

A continuación se va a comprobar que los resultados de las simulaciones se corresponden con el comportamiento de un HEMT con red térmica.

ANÁLISIS DE CONTINUA

Como resultado de este análisis se obtienen las curvas I/V características del transistor.

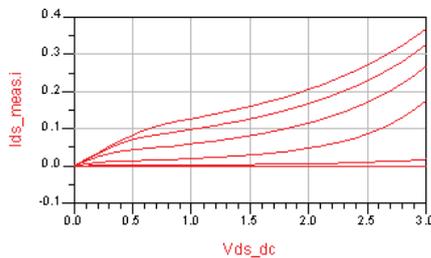


Fig. 7. Curvas I/V del modelo de transistor HEMT creado con Verilog-A

ANÁLISIS DE PARÁMETROS S

Con este análisis se comprueban los parámetros de Scattering S11,S12, S21 y S22 del modelo de transistor que se ha creado con Verilog-A.

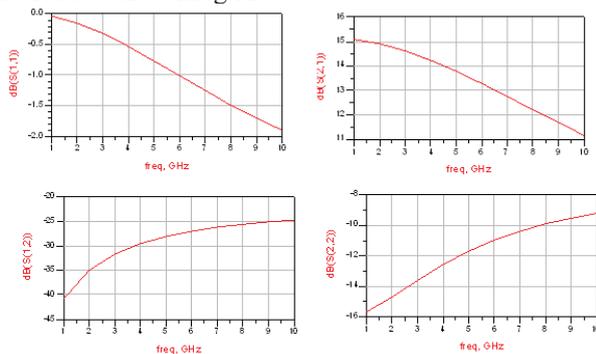


Fig. 8. Parámetros S del modelo de transistor HEMT creado con Verilog-A

IV. CONCLUSIONES

En este documento se han analizado las características principales del lenguaje Verilog-A como herramienta para modelado no lineal de dispositivos.

Verilog-A es un lenguaje de programación que, al ser soportado por diferentes simuladores, permite la creación de modelos para su inclusión en distintas plataformas software, evitando el problema de tener que conocer los rasgos distintivos de cada simulador para introducir nuevos modelos de dispositivos.

AGRADECIMIENTOS

Los autores quieren hacer público su agradecimiento a los siguientes proyectos, dentro de los cuales se han obtenido los resultados presentados en este artículo:

- NoE TARGET (IST program of the EU under contract IST-1-707893-NOE).
- TEC2005-07985-C03-01, Ministerio de Educación y Ciencia

REFERENCIAS

- [1] Simulador Advanced Design System 2004A
- [2] Uso de Verilog-A en ADS (Documentación de ADS)
- [3] Manual de referencia de Verilog-A-ADS (Documentación de ADS)
- [4] Manual de referencia de lenguaje de Verilog-A (Open Verilog Internacional)
- [5] www.tiburon-da.com
- [6] www.agilent.com/find/eesof