

Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Unidade Acadêmica de Engenharia Elétrica

Estágio Integrado

Validação de uma medida de cobertura para a qualificação funcional de circuitos integrados analógicos e mistos utilizando SystemC-AMS

Diego Buriti Araújo

Campina Grande, julho de 2010

Validação de uma medida de cobertura para a qualificação funcional de circuitos integrados analógicos e mistos utilizando SystemC-AMS

Estágio Integrado submetido à Unidade Acadêmica de Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para obtenção da graduação em Engenharia Elétrica.

Aluno:

Diego Buriti Araújo

Orientador:

Raimundo Carlos Silvério Freire

Campina Grande, julho de 2010

Agradecimentos

Primeiramente eu tenho de agradecer ao programa de Intercâmbio BRAFITEC, o qual é promovido pelo Ministério Brasileiro de Educação e Cultura (MEC – CAPES) e os ministérios franceses (MAE – MENESR – DREIC), cuja coordenação é feita pela CDEFI.

Agradeço também à Universidade Federal de Campina Grande e à *École Nationale Supérieure des Systèmes Avancés et Réseaux (ESISAR)* que foram os estabelecimentos responsáveis pelo intercâmbio. Agradeço também aos professores que acreditaram no meu potencial e me selecionaram para participar deste projeto: Raimundo Carlos Silvério Freire e Glauco Fontgalland, e também aos professores que me acompanharam durante a realização desse estágio: Vincent Beroulle e Gui Dehay.

Agradeço aos meus amigos Gilles Fritz e Miguel Augusto de Souza Falcão por todo o suporte e encorajamento que me deram durante o desenvolvimento do meu trabalho.

De nível pessoal, eu agradeço à minha família e todos os meus amigos que estavam no Brasil no tempo que eu estive estagiando, os quais estavam sempre em contato e me encorajando. Agradeço aos meus amigos brasileiros Rodrigo Albuquerque Frazão, Cynthia Bispo Magalhães Pessôa, Gilcianne Andreza Fernandes Araújo e Miguel Augusto de Souza Falcão, os quais estiveram comigo durante toda a minha estadia na França, os quais sempre me apoiaram e deram toda a atenção necessária para que eu conseguisse realizar os meus objetivos acadêmicos.

Things should be as simple as possible, but not simpler.

Albert Einstein

Resumo

Este estágio se encaixa no contexto da verificação de sistemas analógicos e mistos. Ele visa, particularmente, o problema do EVM nos sistemas de comunicação e estuda o impacto da cobertura dos parâmetros do sistema sobre o EVM. A implementação e a aplicação de estímulos necessários para se assegurar a cobertura dos parâmetros foram feitas pela criação de um sistema autônomo de simulação.

A análise do impacto da cobertura dos parâmetros em relação ao valor máximo do EVM foi verificado, gerando estímulos apropriados os quais foram aplicados no sistema com o intuito de diminuir o tempo de teste o qual ele é submetido.

Palavras-chave : SystemC-AMS, EVM, Verificação funcional.

Abstract

Looking through the test of Analog and Mixed systems, especially for the EVM problematic, it was necessary studying the correlation between the system parameters coverage and the EVM. The implementation and the application of stimuli have been done for the creation of an autonomic system. This system stops the simulation when EVM does not increase.

The impact analysis of the parameters coverage in relation with the maximum value of the EVM was made. It's allowing that suitable stimuli for the system test would be applied, decreasing the production time.

Key words: SystemC-AMS, EVM, Functional Verification.

Sumário

Agradecimentos	III
Resumo.....	IV
Abstract	IV
Lista de Figuras.....	VI
Lista de Abreviaturas.....	VII
1 Introdução.....	1
2 Simulação de sistemas analógicos e mistos.....	2
2.1 Necessidades	2
2.2 SystemC-AMS.....	2
2.2.1 Definição	2
2.2.2 <i>SystemC-AMS</i> vs. outras <i>HDL's-AMS</i>	3
2.3 Modelagem e Simulação de Circuitos e Sistemas	5
2.4 Estrutura do SystemC-AMS.....	6
3 Modulador QAM (<i>Quadrature Amplitude Modulator</i>) de 4 bits.....	8
3.1 Modulação.....	10
3.2 Demodulação.....	11
3.2.1 <i>EVM (Error Vector Magnitude)</i>	11
4 Verificação de Sistemas de Comunicação	12
4.1 Objetivos.....	12
4.2 Arquitetura utilizada.....	12
4.3 Resultados.....	14
4.4 Problemas de utilização do SystemC-AMS	20
5 Conclusão	22
6 Perspectivas	23
7 Referências Bibliográficas	24

Lista de Figuras

Figura 1 - Posicionamento das Linguagens de Descrição de Hardware	2
Figura 2 - Posicionamento de SystemC-AMS em relação a outras HDL's [3]	3
Figura 3 - Ilustração de um <i>testbench</i> de um sistema misto	5
Figura 4 - Tipos de sinais suportados por SystemC-AMS	6
Figura 5 –Estrutura geral do SystemC-AMS [5]	6
Figura 6 - Estrutura SystemC-AMS de um sistema elétrico	7
Figura 7 - Gráfico de dois sinais em quadratura	8
Figura 8 - Esquema de um Modulador QAM.....	9
Figura 9 - Tipos de Constelação possíveis de um modulador QAM de 4 bits	10
Figura 10 - <i>EVM (Error Vector Magnitude)</i>	11
Figura 11 - Sistema de parada da simulação.....	13
Figura 12 - Estímulos Utilizados	13
Figura 13 - Sinais interferência inter-símbolo [14].....	13
Figura 14 - Sinais com interferência inter-símbolo [14].....	14
Figura 15 - Evolução da cobertura dos parâmetros em função do tempo	15
Figura 16 - Evolução do <i>EVM</i> em relação ao tempo	16
Figura 17 - Zoom na mudança de valor da evolução do <i>EVM</i> em relação ao tempo	16
Figura 18 – Evolução do <i>EVM</i> em relação à cobertura de parâmetros do sistema	17
Figura 19 - Valores do <i>EVM</i> em relação à uma fraca cobertura dos parâmetros (entrada entre 0 e 3)	17
Figura 20 - Valores do <i>EVM</i> em relação à uma cobertura dos parâmetros (entrada entre 0 e 11)	18
Figura 21 - Valores do <i>EVM</i> em relação à uma cobertura dos parâmetros (entrada entre 0 e 7).....	18
Figura 22 - Valores do <i>EVM</i> em relação à uma cobertura dos parâmetros (entrada entre 0 e 15)	18
Figura 23 - Valores do <i>EVM</i> em relação à cobertura dos parâmetros (do mais fraco ao mais forte)...	19
Figura 24 – Valores do <i>EVM</i> em relação à cobertura dos parâmetros do sistema.....	19
Figura 25 - Densidade espectral de potência	20
Figura 26 - Diferentes escolhas da frequência de amostragem: (a) $f_e < 2f_s$; (b) $f_e = 2f_s$; (c) $f_e > 2f_s$	21

Lista de Abreviaturas

AMS	Analogic and Mixed Systems
BRAFITEC	Brasil / France Ingénieur Technologie
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CDEFI	Conférence des Directeurs des Écoles Françaises d'Ingénieurs
COSY	Systèmes Complexes Coopérants
CTSYS	Conception et Test de Systèmes embarqués
DREIC	Direction des Relations Européennes et Internationales et de la Coopération
ESISAR	l'École Nationale Supérieure des Systèmes Avancés et Réseaux
EVM	Error Vector Magnitude
HDL	Hardware Description Language
HW	Hardware
LCIS	Laboratoire de Conception et d'Intégration des Systèmes
LEN	Linear Electrical Network
MACSY	Modélisation, Analyse et Commande des Systèmes dynamiques
MAE	Ministère des Affaires Étrangères
MEC	Ministério de Educação e Cultura
MENESR	Ministère de l'Éducation et l'Enseignement Supérieur et de la Recherche
ORSYS	Systèmes Optoélectroniques et Radiofréquences
OSCI	Open SystemC Initiative
QAM	Quadrature Amplitude Modulator
RF	Radio Frequency
RFID	Radio Frequency Identification
SDF	Signal Data Flow
SW	Software
UFCG	Universidade Federal de Campina Grande
UWB	Ultra Wide Band
WMS	Wave Mixed Signals

Apresentação da Empresa

O *Laboratoire de Conception et d'Intégration des Systèmes (LCIS)*, fundado em outubro de 1996, foi reconhecido como equipe de formação desde janeiro de 2003 (EA 3747). Ele se trata de uma unidade própria de pesquisa do *Grenoble INP*, igualmente associado à *l'Université Pierre Mendès France*.

O *LCIS* foi o primeiro laboratório de pesquisa universitária em Valence, situado na *École Supérieure d'Ingénieurs En Systèmes Industriels Avancés Rhône-Alpes (ESISAR/INPG)*. Sua atividade de pesquisa é orientada para sistemas embarcados comunicantes. Esta atividade é conjunta com as especialidades de formação da *ESISAR* que, com diferentes especialidades, são representadas pelos professores e pesquisadores efetuando suas atividades de pesquisa no *LCIS*. Os temas abordados incluem a especificação, modelagem, concepção, comunicação, validação e o diagnóstico destes problemas. Nesses diferentes temas, o laboratório propõe novos métodos e modelos, elaborando também ferramentas que são aplicadas ao ambiente da empresa.

A estrutura de pesquisa é dividida nessas quatro áreas:

- **Modélisation, analyse et commande des systèmes dynamiques (MACSY) :**
Os trabalhos têm principalmente o objetivo de desenvolver leis seguras de comando e/ou supervisão de alta performance dos sistemas dinâmicos e distribuídos.
- **Systèmes complexe coopérants (COSY) :**
O objetivo é principalmente propor um quadro e um formalismo para a engenharia de sistemas complexos abertos baseados no conceito de um agente. A parte é provida de modelos, arquiteturas, etapas de análise/concepção, de validação e de simulação específicas com a finalidade de atender às necessidades de especificação de aplicações mais críticas (sistemas físicos reais, serviços de internet).
- **Systèmes optoélectroniques et radiofréquence:**
Os trabalhos desenvolvidos neste tema correspondem os projetos MESSYS e ORSYS. Eles se articulam em torno da utilização de tecnologias e sinais rádiofrequência e óticos para a comunicação, tratamento dos sinais e medição. As pesquisas englobam o desenvolvimento de sistemas integrados, sistemas do tipo *Wireless*, os quais compõem as etiquetas *RFID* e até o conjuntos *UWB*. Os trabalhos se desenvolvem em

torno da modelagem e simulação eletromagnética, a concepção e otimização de sistemas mistos, a medição e caracterização experimental.

- **Conception et teste de systèmes embarqués (CTSYS) :**

As atividades de pesquisa desenvolvidas dentro do tema “Confiabilidade de Funcionamento” são relativas à concepção dos sistemas embarcados e distribuídas com foco neste aspecto. A atividade de teste tem como objetivo analisar o mais detalhado possível as funcionalidades e/ou a estrutura de cada seção do sistema e detectar o maior número de defeitos possível.

Validação de uma medida de cobertura para a qualificação funcional dos circuitos integrados analógicos e mistos utilizando SystemC-AMS

1 Introdução

Atualmente, os circuitos integrados estão presentes em inúmeros produtos, podendo ocasionalmente apresentar erros de concepção, os quais são susceptíveis a erros de funcionamento. Os circuitos são submetidos a vários testes com finalidade de verificar se seu funcionamento está correto. Para ser considerado um bom circuito, ele deve respeitar às especificações iniciais que contém as funções realizadas por cada bloco do sistema, as limitações físicas em amplitude e em frequência dos sinais, etc.

Antes de sua fabricação, os circuitos são submetidos à testes funcionais para verificar suas funcionalidades, fazendo variar os parâmetros e os sinais dos estímulos de entrada, observando as respostas geradas na saída e comparando-as com os valores esperados. Deve-se assegurar que para todas as possibilidades de estímulos aplicados, a cobertura de todas as variações dos sinais do circuito sejam realizadas com o mínimo de erro possível na saída.

O circuito é desenvolvido para atender as características das especificações iniciais com uma certa tolerância sobre seus parâmetros (intervalo entre os valores mínimos e máximos). Um tipo de medida de qualidade da verificação funcional consiste em variar um destes parâmetros fora de seu intervalo para verificar a resposta do sistema. Tal resposta não será necessariamente falsa, podendo ser apenas uma falha do ambiente de verificação.

Um bom teste funcional é aquele que cobre todas as possibilidades de erros em um sistema integrado. Para realizá-lo, é necessário o uso de uma linguagem de descrição de hardware (*HDL*) que permita criar diversos componentes de um sistema, individualmente, e gerar estímulos na entrada para que a saída do sistema seja estudada e comparada com a resposta desejada (teórica). A linguagem de descrição de hardware SystemC-AMS permite obter uma simulação de alto nível de abstração. Ela foi desenvolvida pela *OSCI* (Open SystemC Initiative) [1] [2], uma linguagem gratuita que permite simular diferentes componentes (analógicos e mistos) ou programas de alto nível.

2 Simulação de sistemas analógicos e mistos

2.1 Necessidades

Devido à necessidade de encontrar soluções para diferentes problemas de aplicação, a complexidade dos sistemas e circuitos eletrônicos aumentou. Tornou-se necessário desenvolver novos ambientes que nos permitissem realizar simulações a fim de obter-se um bom nível de qualidade em relação às suas funcionalidades.

Simulando os sistemas e aplicando faltas artificiais, variando todos os parâmetros do sistema fora de seus intervalos permitidos, assegura-se que os testes realizados cobrem todos os erros possíveis.

2.2 SystemC-AMS

2.2.1 Definição

SystemC-AMS (SystemC – Analogic Mixed Systems) é uma linguagem de programação desenvolvida pela *OSCI (Open SystemC Initiative)* [1] [2] que permite realizar simulações de sistemas analógicos e mistos em alto nível de abstração em relação às outras linguagens tais como: *VHDL-AMS* e *Verilog-AMS*. Esta programação de alto nível diminui o tempo de simulação consideravelmente. *SystemC-AMS* é uma biblioteca C++ que permite trabalhar com sinais analógicos (tempo contínuo) e digitais (discretos).

Os sistemas *HW* (hardware) e *SW* (software) podem ser simulados em *SystemC-AMS*. Em inúmeros casos, os sistemas *SW* interagem com a parte analógica do sistema. A simulação das partes digitais devem ser feitas em conjunto com as analógicas devido à influência das mesmas nos componentes do sistema.

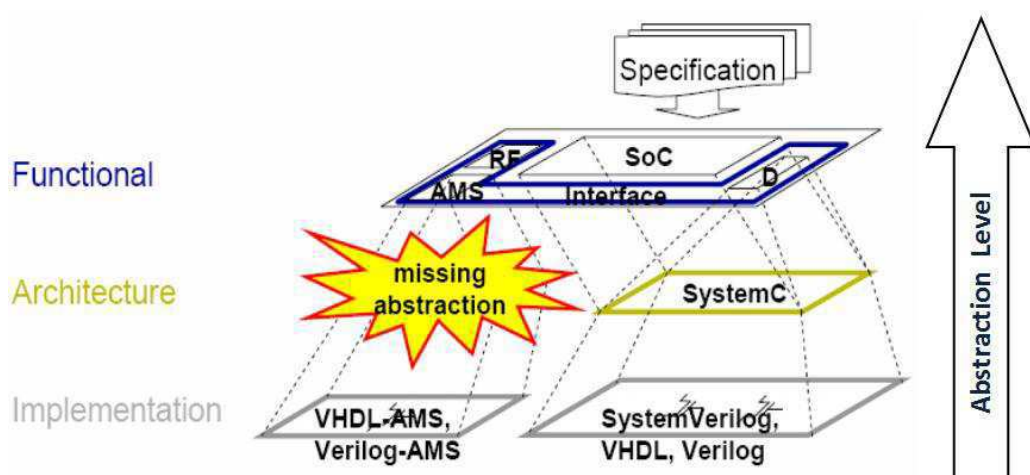


Figura 1 - Posicionamento das Linguagens de Descrição de Hardware

A Figura 1 ilustra o posicionamento de *SystemC-AMS* e das outras linguagens de descrição de hardware, mostrando a predominância de cada linguagem em diferentes domínios nos quais cada linguagem predomina.

O objetivo de *SystemC-AMS* é completar as funcionalidades de *SystemC* (orientado HW/SW) oferecendo as seguintes possibilidades para os sistemas analógicos e mistos :

- Modelagem funcional ;
- Exploração da arquitetura ;
- Integração do sistema ;
- Verificação do sistema ;
- Prototipagem virtual.

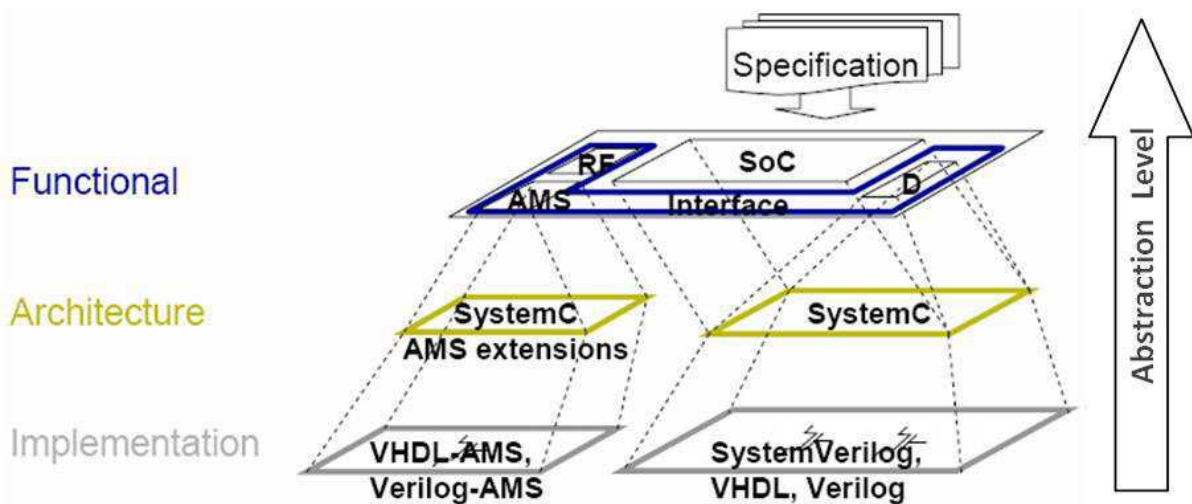


Figura 2 - Posicionamento de *SystemC-AMS* em relação a outras HDL's [3]

2.2.2 *SystemC-AMS* vs. outras HDL's-AMS

A simulação dos sistemas analógicos e mistos antes da fabricação é uma etapa muito importante da produção de sistemas integrados. Várias linguagens de programação permitem a simulação destes circuitos, cada um com propriedades e características diferentes que podem, ou não, facilitar esta simulação em função do tipo de sistema e de sua complexidade.

As HDL's mais utilizadas são o *VHDL-AMS* e o *Verilog-AMS*, porém, o *SystemC-AMS* começou a ser utilizado para realizar verificações funcionais de circuitos integrados analógicos e mistos. As HDL's foram criadas para atender às necessidades do processo de concepção do circuito integrado, e principalmente da verificação funcional. Em seguida consta uma breve lista destas necessidades:

- Permitir a descrição da arquitetura de um sistema que é decomposto em sub-sistemas interconectados ;
- Especificar a função de um sistema utilizando uma linguagem de programação;
- Simular o comportamento do sistema antes da fabricação para realizar possíveis modificações antes da produção ;
- Dar acesso à arquitetura detalhada de um sistema para que o mesmo seja sintetizado a partir de uma especificação mais precisa, permitindo a diminuição do tempo de mercado.

A Tabela1 mostra as diferenças entre *VHDL-AMS*, *Verilog-AMS* e *SystemC-AMS*:

Tabela 1 - Comparação entre VHDL-AMS [4], Verilog-AMS [4] e SystemC-AMS

	Características	VHDL-AMS	Verilog-AMS	SystemC-AMS
Aspectos da linguagem	Herança	Ada	C	C / C++
		<i>Case sensitive</i>		
	Modulação	Separação de sinais externos (<i>entitys</i>) e internas (<i>architectures</i>)	Os módulos incluem os sinais internos e externos	Separação dos sinais externos (<i>modules</i>) e internos
	Bibliotecas	Sim (pré-compiladas)	Não	Sim (pré-compiladas)
Expressões da estrutura	Portas	Contínuas e discretas		
		Conservativas e não-conservativas (<i>signal-flow</i>)		
		Portas contínuas não são modeladas	Sim (do tipo <i>inout</i>)	Sim (do tipo <i>inout</i>)
	Composição	Instanciação hierárquica dos componentes		
Descrição do comportamento	Objetos	Terminais, quantidades, sinais, variáveis, constantes	Variáveis, fios, flip-flops	Terminais, Sinais, variáveis, constantes
	Condições	Simultânea, sequencial, contínua (simultânea e procedural)		
		As condições contínuas podem ser misturadas com as condições instantâneas.	As condições precisam ser agrupadas em blocos analógicos. Um bloco analógico por módulo.	As condições contínuas podem ser misturadas com as condições instantâneas.
Aspectos dos sinais mistos	Interação comportamental	Acesso dos sinais discretos em um contexto contínuo	Acesso das redes discretas e variáveis em um contexto contínuo	Acesso dos sinais discretos em um contexto contínuo
		Acesso das quantidades contínuas em um contexto discreto	Acesso das redes discretas e variáveis em um contexto discreto	Acesso das quantidades contínuas em um contexto discreto

2.3 Modelagem e Simulação de Circuitos e Sistemas

A modelagem está situada no núcleo de qualquer processo de concepção. Esta tarefa consiste essencialmente em desenvolver as descrições abstratas de uma certa realidade física de tal maneira que elas sejam úteis para o processo de concepção. Os sistemas mistos possuem diferentes tipos de sinais tratados; os tratados pelos sistemas analógicos são de tempo contínuo e os tratados pelos sistemas digitais são discretos.

Os módulos digitais são modelados por processos discretos que são ativados por *clocks* (sincronizados). Tais processos podem ser utilizados para modelar sistemas desde o de transistores até o nível algorítmico (próximo do software).

Os módulos analógicos são especificados por equações diferenciais ou algébricas. Os sinais analógicos mudam de valores continuamente e podem ser classificados como conservativos ou não-conservativos. Os sinais conservativos são bi-direcionais e representam uma quantidade “*across*” e “*through*” (tensão e corrente elétricas, respectivamente). Eles devem respeitar as leis da física, como por exemplo, as leis de Kirchoff.

Todos os módulos criados possuem uma extensão “.h” (*Header File*) que já se encontra incluída na função “*main()*”. Pode-se reutilizar módulos já criados para desenvolver outros blocos do sistemas que necessitam dos mesmos.

Uma simulação em SystemC-AMS é realizada a partir de módulos que correspondem, geralmente, aos componentes do sistema. Pode-se também criar um módulo que representa um conjunto de componentes ou um sub-sistema. Os módulos produzidos são referenciados por objetos instanciados na função “*main()*” e as ligações entre eles são realizadas nesta mesma função. A simulação de um circuito é feita com um *testbench*, o qual é um processo que permite gerar sinais nas entradas do sistema, permitindo comparar as saídas com os valores de referência (ideais), como mostrado na Figura 3:

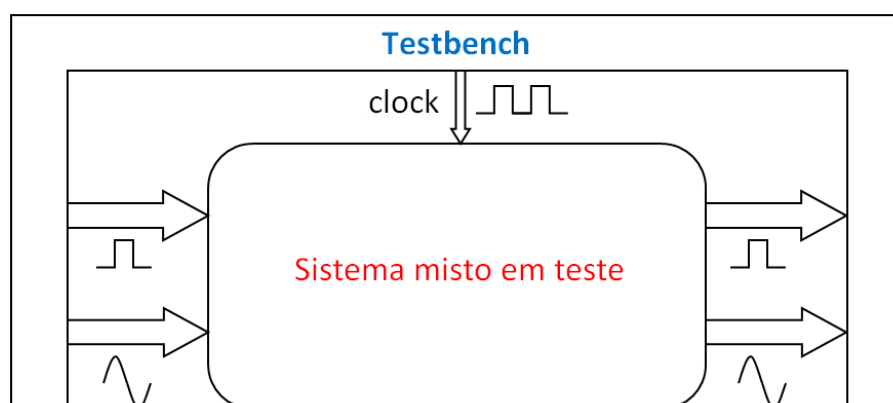


Figura 3 - Ilustração de um *testbench* de um sistema misto

SystemC-AMS trabalha com sinais discretos, ou seja, ele necessita amostrar os sinais analógicos para processá-los de forma discreta. Tais formas dos sinais são ilustradas na Figura 4:

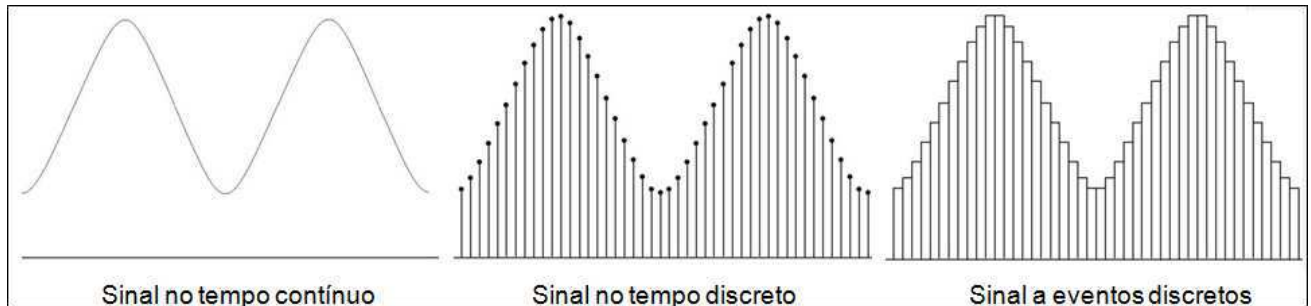


Figura 4 - Tipos de sinais suportados por SystemC-AMS

2.4 Estrutura do SystemC-AMS

SystemC-AMS trabalha com sinais analógicos e mistos, logo é necessária uma sincronização entre as diferentes partes analógicas do sistema para que os dados sejam processados. A Figura 5 mostra a estrutura do *SystemC-AMS*.

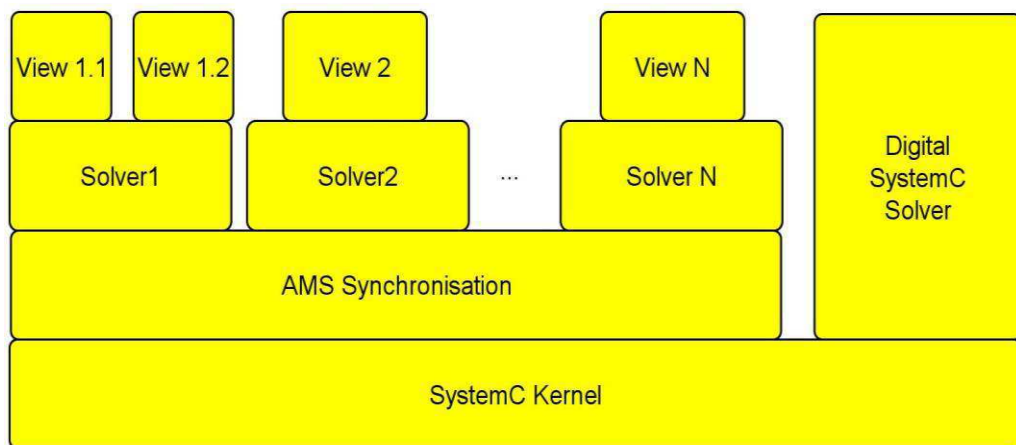


Figura 5 –Estrutura geral do SystemC-AMS [5]

Observando a figura acima, percebe-se que existe uma camada de sincronização entre os diferentes *solvers* do sistema que permite trabalhar com sinais analógicos de diversas frequências, fazendo suas amostragens para que o núcleo (*SystemC kernel*) possa processar ambos os sinais (analógicos e digitais). A seguir, são detalhadas os diversos blocos funcionais da estrutura do *SystemC-AMS*.

- *Views* – descrições do utilizador sob forma de equações, como por exemplo, a criação das equações de um circuito elétrico, utilizando as leis de Kirchoff; [6]
- *Solvers* – resolvem as diferentes equações criadas pelas *Views*.
- *AMS Synchronisation* – esta camada faz a interconexão entre os *solvers* e o *SystemC Kernel*.

A Figura 6 exemplifica a estrutura *SystemC-AMS* de um sistema misto:

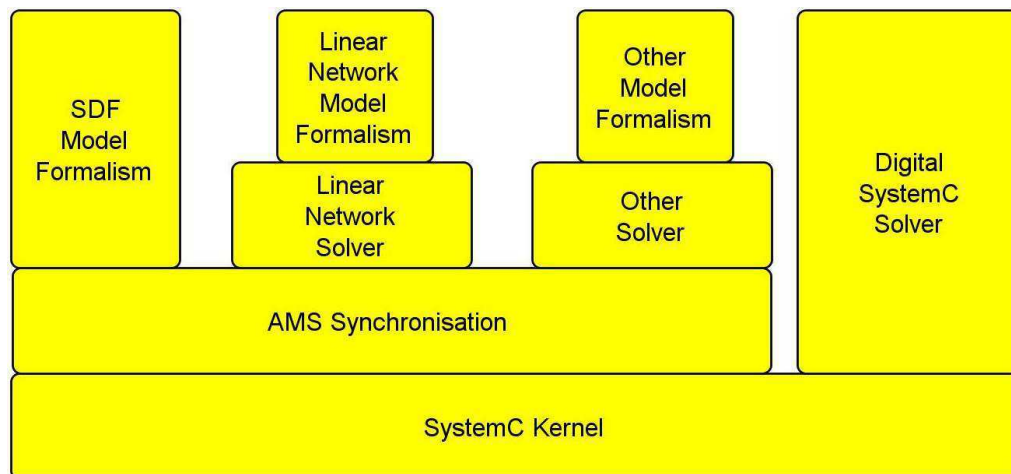


Figura 6 - Estrutura *SystemC-AMS* de um sistema elétrico

Observando a Figura 6, percebe-se a presença de um modelo de formalismo *SDF* (*System Data Flow*). Abaixo, estão listadas as propriedades do domínio *SystemC-AMS SDF*:

- As amostras *SDF* são associadas a SC_TIME [7];
- Os sinais a eventos discretos são amostrados a um tempo Δ_0 fixo (período de amostragem) [7];
- O período de amostragem se propaga através de todo o sistema. Desta maneira, ele é definido por um módulo, o qual deverá ser o mesmo para todo o sistema [7].

Nota-se a presença de um modelo de formalismo de uma Rede Elétrica Linear. A seguir são descritas as propriedades do domínio *SystemC-AMS LEN* (*Linear Electrical Networks*).

- Será associado um sistema de equações para cada instância da rede conectada [7];
- Todas as redes necessitam ser conectadas, ao menos, a um nó de referência [7];
- A rede será re-inicializada assim que um dos elementos da matriz mude de valor, como por exemplo, uma resistência [7];
- Todas as conexões com o domínio *SDF* devem ter a mesma taxa de amostragem [7];
- Os elementos das redes são pré-definidos [7].

3 Modulador QAM (*Quadrature Amplitude Modulator*) de 4 bits

Um modulador QAM tem a função de enviar informações digitais através de duas vias de transmissão analógicas com uma portadora diferente para cada uma dessas vias. As portadoras utilizadas pelo modulador QAM são funções Seno e Cosseno, pois elas possuem uma defasagem de 90° entre si (quadratura).

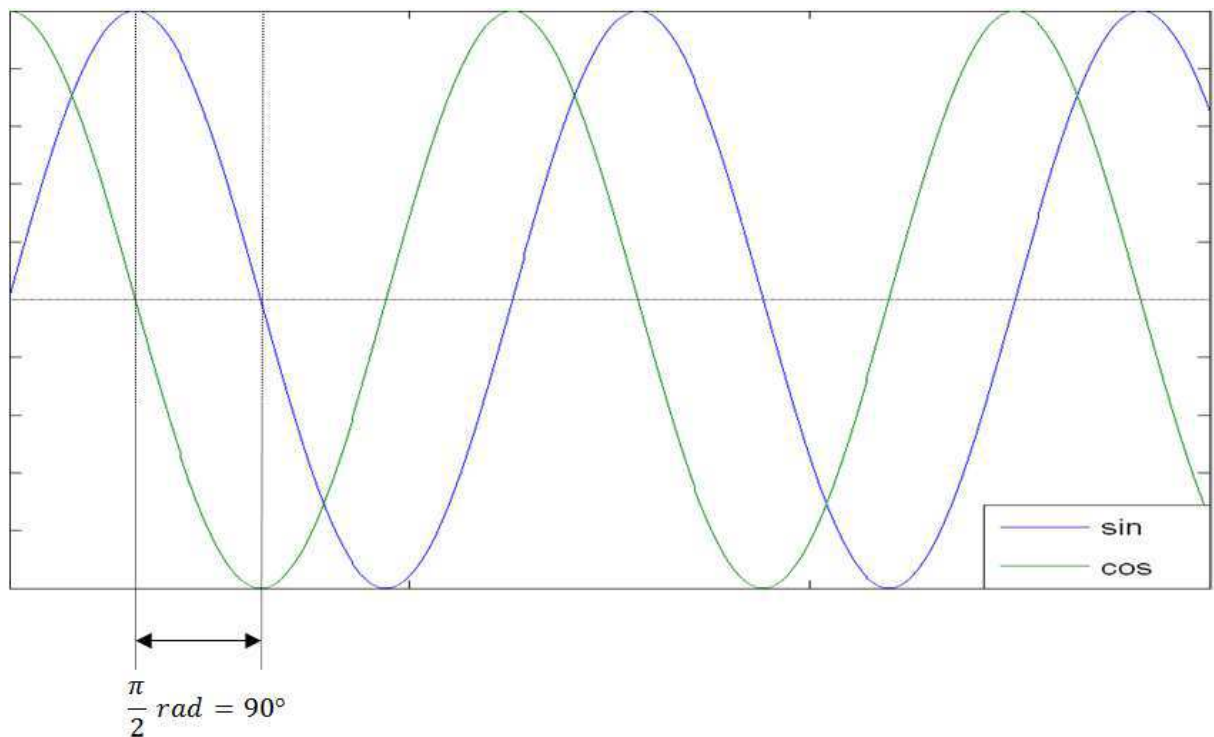


Figura 7 - Gráfico de dois sinais em quadratura

Um esquema representativo de um modulador QAM é mostrado na Figura 8, onde são mostrados todos blocos funcionais assim como são indicados os parâmetros do sistema (sinais digitais e analógicos) e em seguida são deduzidas as expressões de funcionamento.

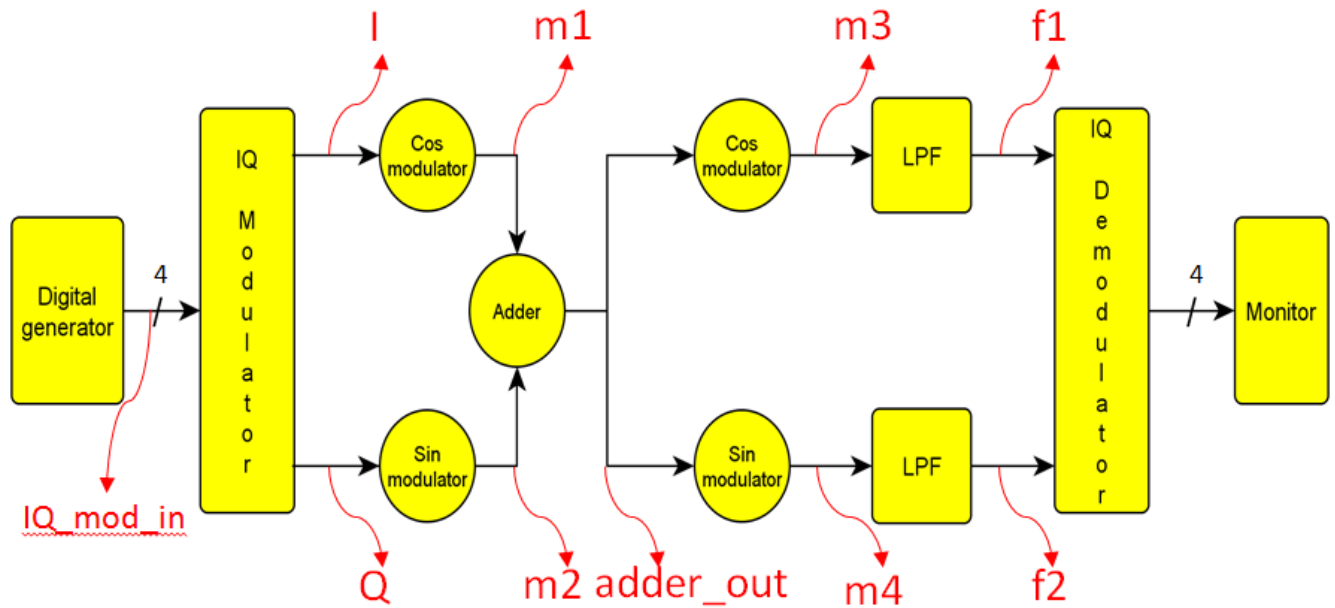


Figura 8 - Esquema de um Modulador QAM

seja :

$$m1 = I \times \text{Cos}(\omega t)$$

$$m2 = Q \times \text{Sin}(\omega t)$$

agora, teremos :

$$adder_out = [(I \times \text{Cos}(\omega t)) + (Q \times \text{Sin}(\omega t))]$$

ou seja:

$$m3 = [I \times \text{Cos}(\omega t) + Q \times \text{Sin}(\omega t)] \times \text{Cos}(\omega t) = I \times \text{Cos}^2(\omega t) + Q \times \text{Sin}(\omega t) \times \text{Cos}(\omega t)$$

$$m3 = \left[I \times \frac{1}{2} \times (\text{Cos}(\omega t - \omega t) + \text{Cos}(\omega t + \omega t)) \right] + \left[Q \times \frac{1}{2} \times \text{Sin}(\omega t + \omega t) \right]$$

$$m3 = \frac{I}{2} + \frac{I}{2} \times \text{Cos}(2\omega t) + \frac{Q}{2} \times \text{Sin}(2\omega t)$$

$$m4 = [I \times \text{Cos}(\omega t) + Q \times \text{Sin}(\omega t)] \times \text{Sin}(\omega t) = Q \times \text{Cos}^2(\omega t) + I \times \text{Sin}(\omega t) \times \text{Cos}(\omega t)$$

$$m4 = \left[Q \times \frac{1}{2} \times (\text{Cos}(\omega t - \omega t) - \text{Cos}(\omega t + \omega t)) \right] + \left[I \times \frac{1}{2} \times \text{Sin}(\omega t + \omega t) \right]$$

$$m4 = \frac{Q}{2} - \frac{Q}{2} \times \text{Cos}(2\omega t) + \frac{I}{2} \times \text{Sin}(2\omega t)$$

Os filtros possuem uma frequência de corte inferior a $2\omega t$, então :

$$m_3 = I \times \frac{1}{2}$$

$$m_4 = Q \times \frac{1}{2}$$

3.1 Modulação

O modulador *QAM* é implementado de maneira que para cada valor de entrada, as saídas (*I* e *Q*) possuam valores que formem uma constelação bem definida. Neste caso, um modulador de 4 bits foi utilizado, e o número de valores que o modulador possui nas saídas é igual a 2^N , onde *N* é o número de bits do modulador.

$$\text{Número de saídas} = 2^N = 2^4 = 16$$

A quantidade de pontos da saída desse modulador é igual a 16. O conjunto das saídas possíveis formam uma constelação bem definida, mas como em todos os sistemas reais, estão presentes os erros de aproximação. Na Figura 9 são mostradas dois tipos de constelação utilizadas pelo modulador *QAM* de 4 bits.

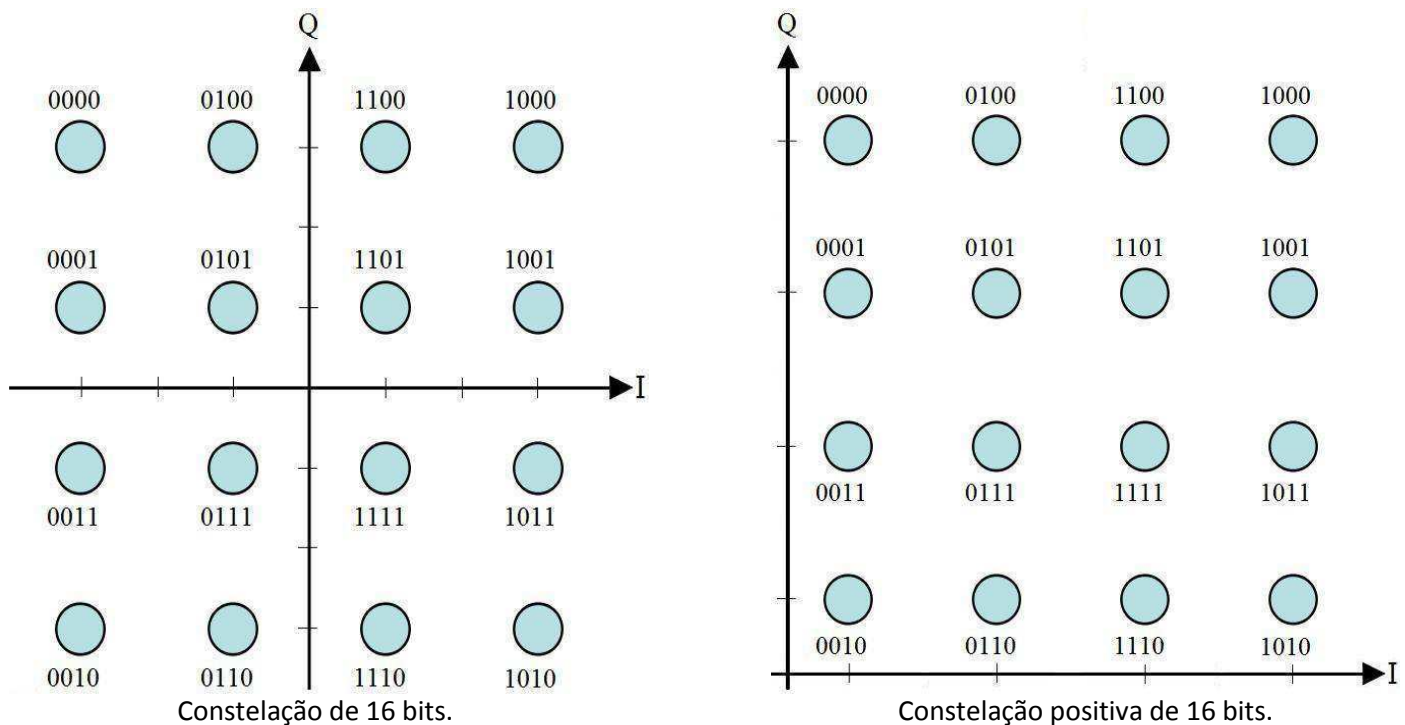


Figura 9 - Tipos de Constelação possíveis de um modulador QAM de 4 bits

Percebe-se que existem dois tipos diferentes de constelação. Uma com valores positivos e negativos os quais necessitam de uma detecção de fase para distinguir os valores de mesmo módulo e sinais opostos, enquanto que a positiva não possui esta necessidade.

3.2 Demodulação

Observando-se a constelação, nota-se que ela possui valores positivos e negativos de mesmo módulo. Para diferenciar tais valores ao nível do demodulador, é necessário um detector de fase.

Uma maneira mais simples de implementar um modulador *QAM* sem a necessidade de um detector de fase consiste em utilizar uma constelação positiva. Esta constelação permite associar os valores correspondentes de *I* e *Q* de cada entrada do sistema.

Diversos erros podem surgir na transmissão como, por exemplo, o ruído de fase, a oscilação residual das saídas dos filtros etc. Tais erros fazem com que a constelação receba valores não ideais, os quais podem provocar falsas conversões de dados em relação ao demodulador ideal.

3.2.1 EVM (Error Vector Magnitude)

O EVM é o erro associado à reconstituição da constelação pelo demodulador segundo os valores recebidos pelo receptor. A cada pulso do *clock*, um módulo que realiza o cálculo do EVM lê os valores de *I* e *Q*, calcula os valores de referência de *I* (I_{ref}) e *Q* (Q_{ref}) para em seguida calcular o erro.

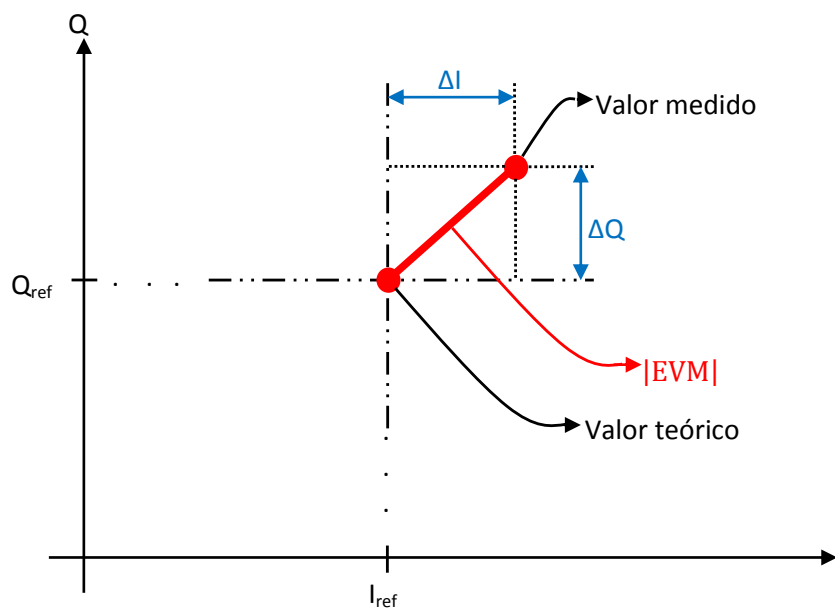


Figura 10 - EVM (Error Vector Magnitude)

4 Verificação de Sistemas de Comunicação

O processo de fabricação dos circuitos analógicos e mistos atuais tornou-se muito caro devido às suas complexidades. Desta forma, a verificação tornou-se uma etapa muito importante neste processo. Esta verificação consiste em fazer um *testbench* que permite achar erros de projeto.

O *testbench* é uma maneira de testar o funcionamento do sistema aplicando sinais com respostas conhecidas na entrada para observar a(s) saída(s) do sistema e compará-las com os valores esperados (ideais). A verificação consiste em achar erros de concepção e de observar a cobertura dos parâmetros do sistema que, no caso, é a variação dos valores máximos e mínimos das amplitudes dos sinais em relação aos valores teóricos das mesmas.

4.1 Objetivos

O objetivo desejado é realizar uma automatização da verificação funcional de um modulador *QAM* estudando o impacto que as coberturas dos intervalos dos vários sinais do sistema possuem sobre o *EVM*. Deseja-se verificar uma correlação entre este erro e as porcentagens de cobertura da variação dos sinais em relação a seus valores teóricos máximos. Quando os sinais possuem variações máximas iguais às teóricas, os erros devem ser máximos.

Aplicam-se diversos estímulos aleatórios, sem que os intervalos dos sinais sejam aumentados. O programa deverá continuar a simulação com outros estímulos aleatórios para que uma melhor cobertura dos intervalos dos sinais seja obtida.

Esta automatização é feita com um critério de parada da simulação quando o *EVM* não evolui mais em relação à cobertura dos parâmetros. A cada subida do *clock*, o programa verifica se o valor atual é superior ao valor precedente. Um contador é incrementado cada vez que este valor é igual ou inferior ao valor precedente, e ajustado para zero quando o valor é superior.

4.2 Arquitetura utilizada

Aplicamos estímulos os quais contém os precedentes. Seja J_1 o primeiro estímulo aplicado ao sistema. A simulação encontrará um valor de erro em função de J_1 . Os valores máximos da variação de cada parâmetro do sistema e o valor do *EVM* são gravados em um arquivo texto cada vez que uma simulação é realizada.

Em seguida, executamos a simulação com um estímulo J_2 , o qual contém J_1 e gravamos os valores gerados pela simulação no mesmo arquivo de texto. O objetivo final é obter uma quantidade de dados provenientes de várias simulações. O esquema do sistema de interrupção da simulação e o de utilização dos estímulos utilizados são ilustrados nas figuras 11 e 12 respectivamente.

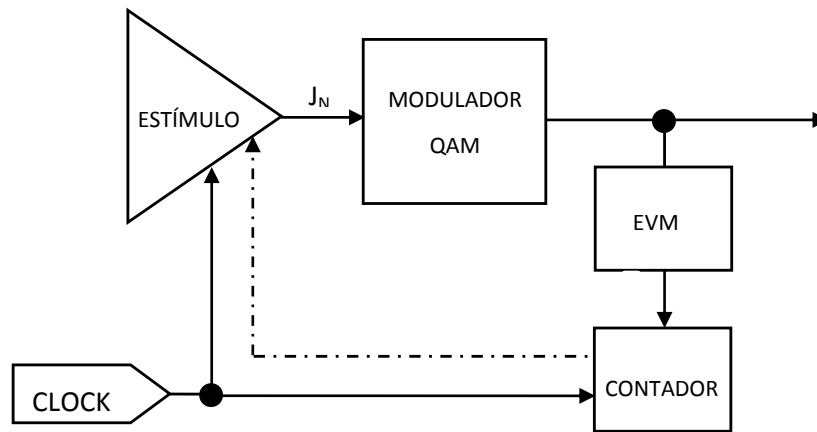


Figura 11 - Sistema de parada da simulação

Observando a figura acima, notamos a presença de um contador cuja função é indicar o simulador para interromper a simulação e de mudar os estímulos de entrada quando o *EVM* não evolui mais. O valor do contador é incrementado cada vez que o valor do *EVM* é igual ou superior ao valor precedente. Ele é ajustado para o valor zero quando um valor superior do *EVM* é encontrado.

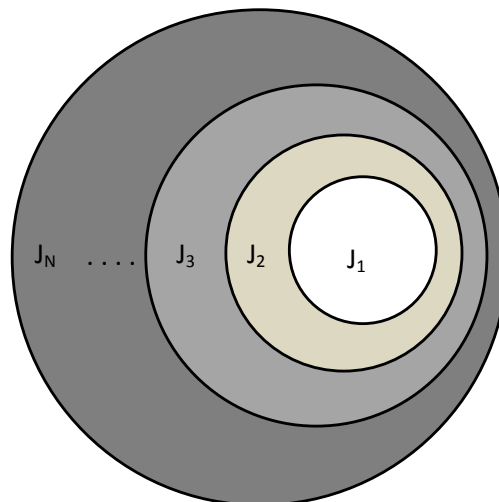


Figura 12 - Estímulos Utilizados

Utilizando esses estímulos na entrada, assegura-se que o *EVM* será maior a cada simulação, pois o conjunto de sinais atual contém o precedente. Obter uma cobertura máxima de todos os parâmetros do sistema é uma condição necessária para a verificação, mas não suficiente. De fato, um dos fatores que intervêm sobre o impacto do erro é a interferência inter-símbolo das entradas.

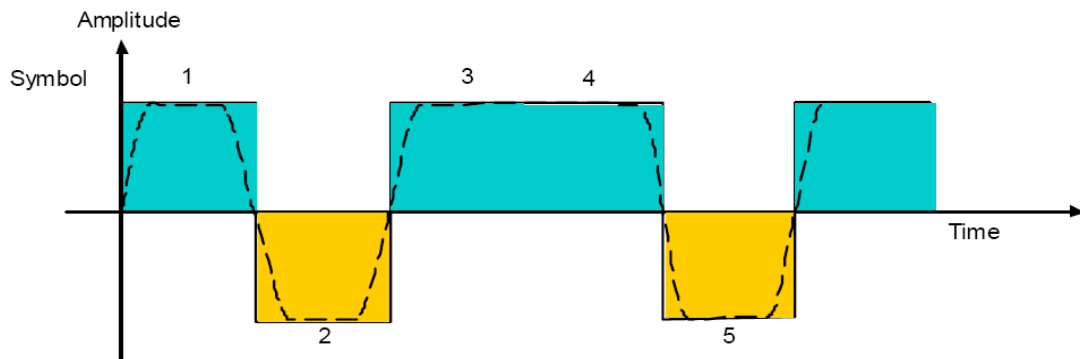


Figura 13 - Sinais sem interferência inter-símbolo [12]

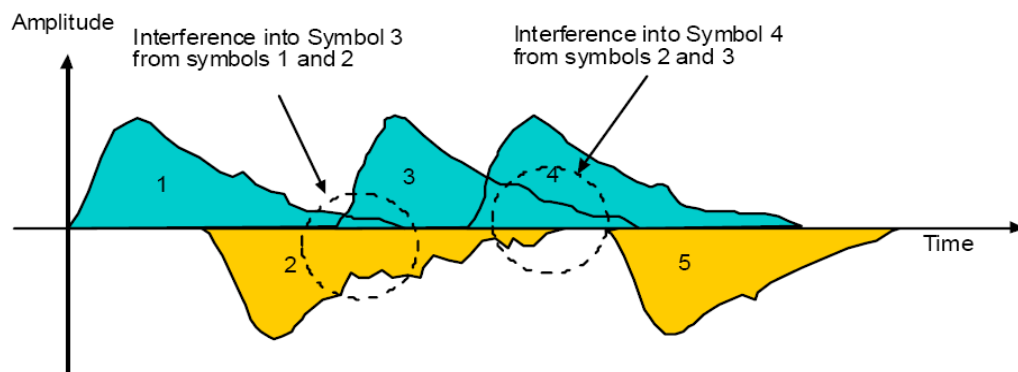


Figura 14 - Sinais com interferência inter-símbolo [12]

4.3 Resultados

O modulador *QAM* desenvolvido gera uma constelação com 16 pontos na saída. É utilizada uma fonte digital na entrada do modulador de frequência 1 kHz que gera valores aleatórios entre 0 e 15. O primeiro ponto consiste em realizar uma simulação com um conjunto de estímulos J_1 , medindo os erros de construção da constelação.

Desconsiderando-se a interferência inter-símbolo apenas como uma simplificação do projeto, o pior caso do erro obtido é quando ocorre uma sequência **mínimo**¹ -> **máximo**² -> **mínimo** nos valores de *I* e *Q*. A duração de uma simulação depende da probabilidade de se ter o pior caso dos valores na entrada os quais geram o maior erro na saída. O tempo de simulação é calculado de acordo com o período do sinal de entrada para que a probabilidade de obter este conjunto de entradas consecutivas.

$$\text{Tempo de simulação}_{\text{mínimo}} = \left(\frac{1}{16} \times \frac{1}{16} \times \frac{1}{16} \right)^{-1} \times \frac{1}{f_{\text{fonte}}}$$

$$\text{Tempo de simulação}_{\text{mínimo}} = 4096 \times T_{\text{fonte}}$$

¹ Valor na constelação onde o valor de *I* e *Q* é mínimo (amplitude igual a 1 ; sinal de entrada igual a 2)

² Valor na constelação onde o valor de *I* e *Q* é máximo (amplitude igual a 4 ; sinal de entrada igual a 12)

Isto significa que é necessário enviar 4096 entradas diferentes para obter uma boa probabilidade de obter a sequência $2 \rightarrow 12 \rightarrow 2$ na entrada para que os valores de I e Q possuam variações e que se obtenha o maior *overshoot* possível na construção da constelação.

Na verificação realizada, o sinal de entrada possui uma frequência de 1 kHz, e como necessita-se enviar ao menos 4096 números na entrada do simulador, adota-se o tempo de simulação de 5 ms (aproximadamente 5000 números enviados) pelo primeiro conjunto de estímulos J_1 . Salva-se os valores de cada parâmetro e do *EVM* máximo obtido nas entradas do demodulador. Em seguida, a simulação é realizada novamente, aplicando-se o conjunto de estímulos J_2 ($J_2 \supset J_1$) e um tempo de simulação duas vezes maior.

De forma geral,

$$\text{Tempo de simulação}_{\text{mínimo}} = N \times J_N$$

Uma vez que várias simulações foram realizadas, inúmeros pontos que representam a evolução do *EVM* e da cobertura dos parâmetros em função do tempo estão salvos. Os comportamentos desses parâmetros são mostrados nas curvas a seguir.

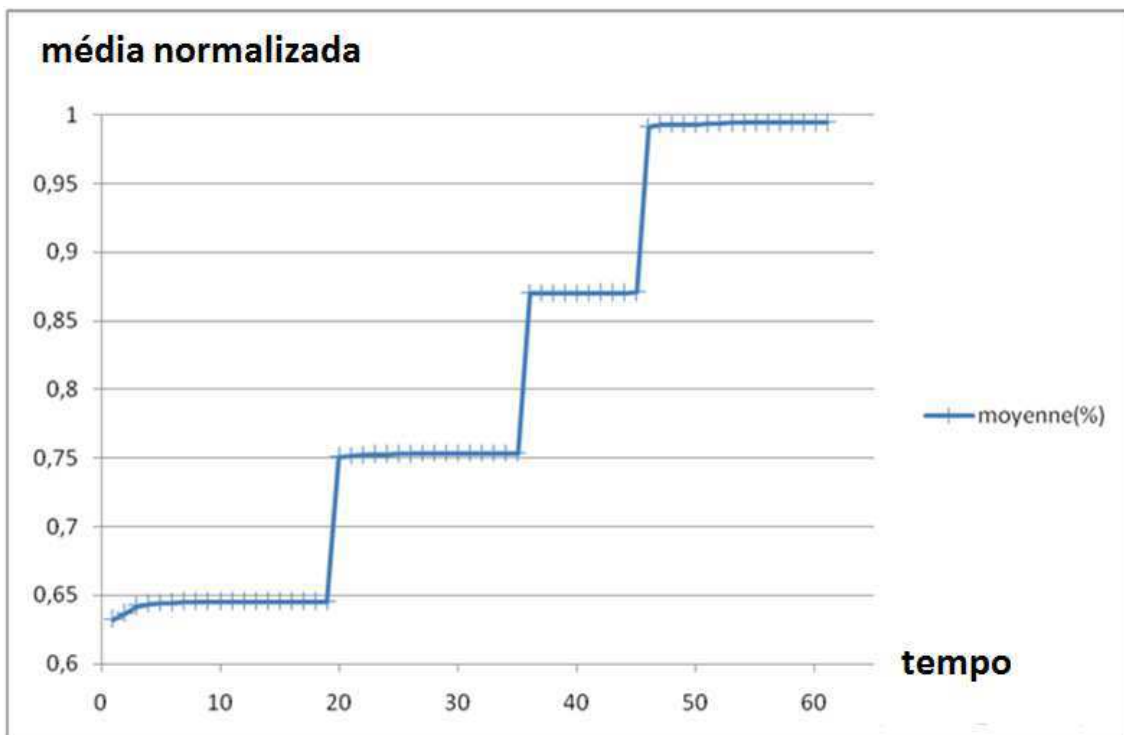


Figura 15 - Evolução da cobertura dos parâmetros em função do tempo

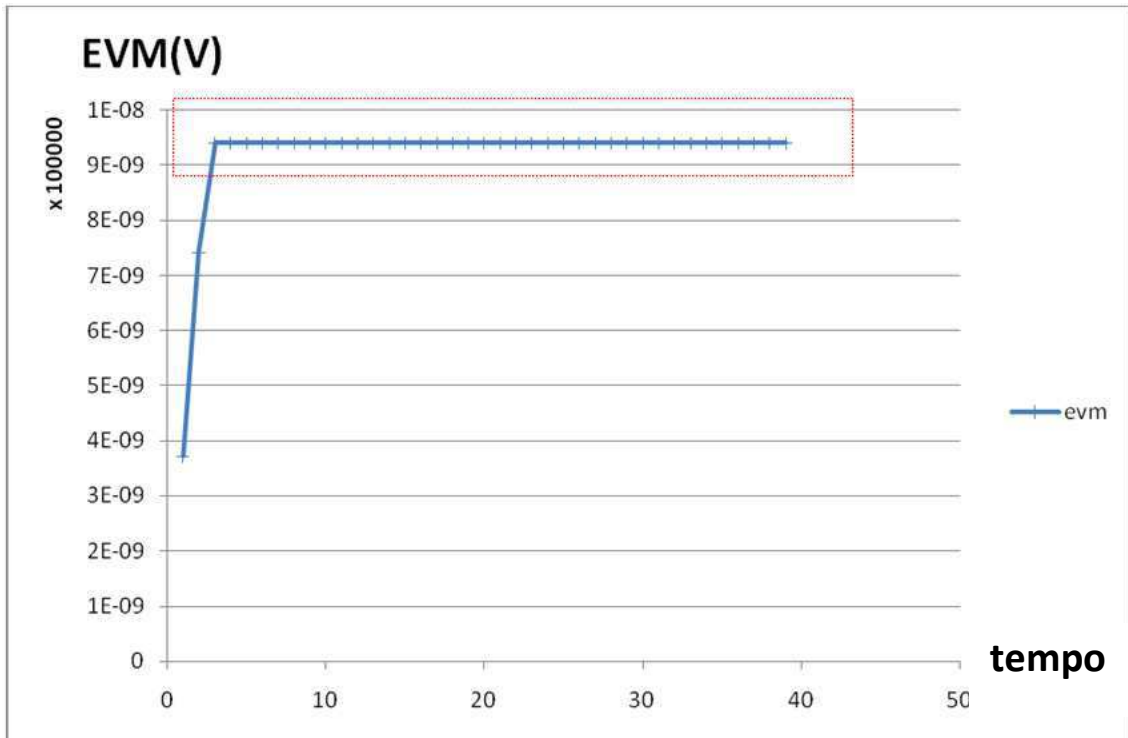


Figura 16 - Evolução do EVM em relação ao tempo

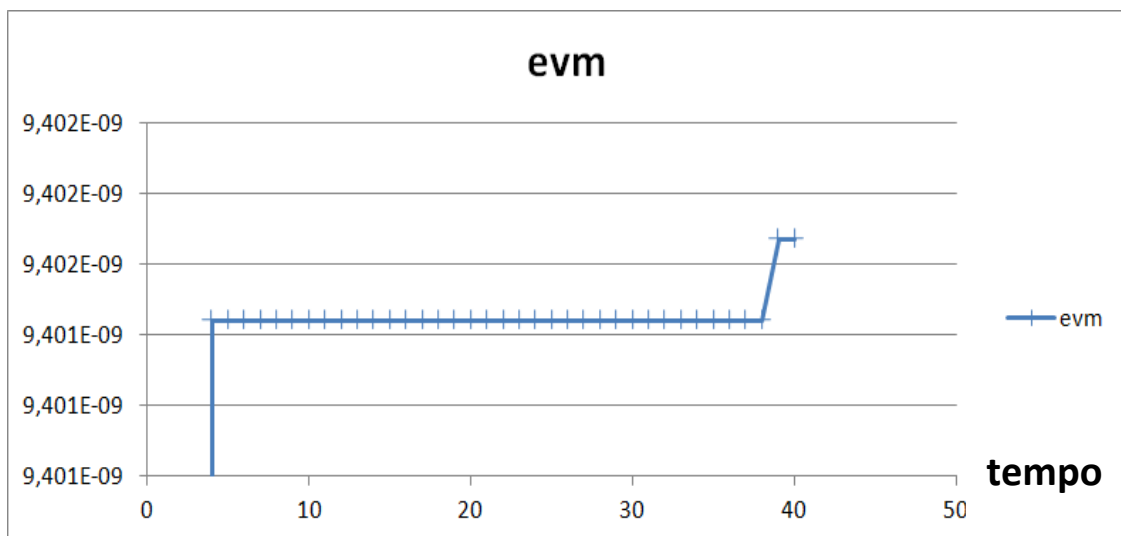


Figura 17 - Zoom na mudança de valor da evolução do EVM em relação ao tempo

Percebe-se que o EVM e a cobertura dos parâmetros obtiveram um comportamento crescente em relação ao tempo, ou seja, pode-se concluir que o EVM está correlacionado com a cobertura dos parâmetros do sistema. Em seguida, nos preocuparemos com a evolução do EVM em relação à cobertura dos parâmetros.

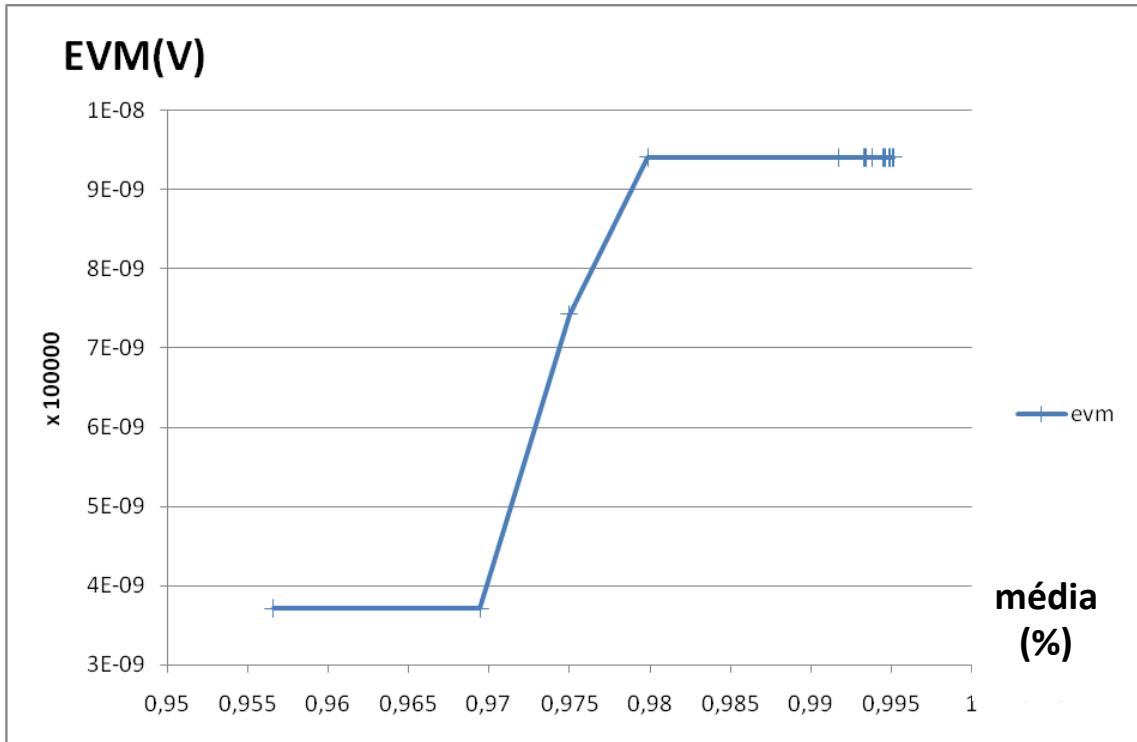


Figura 18 – Evolução do EVM em relação à cobertura de parâmetros do sistema

Para mostrar a correlação entre o *EVM* e a cobertura dos parâmetros do sistema, pode-se aplicar estímulos na entrada cuja cobertura dos parâmetros seja fraca. Pode-se traçar a curva da evolução do *EVM* em relação à cobertura dos parâmetros para verificar esta correlação. Aplicando-se estes estímulos de entrada **[0 ; 3]**, obtemos a curva mostrada pela Figura 19:

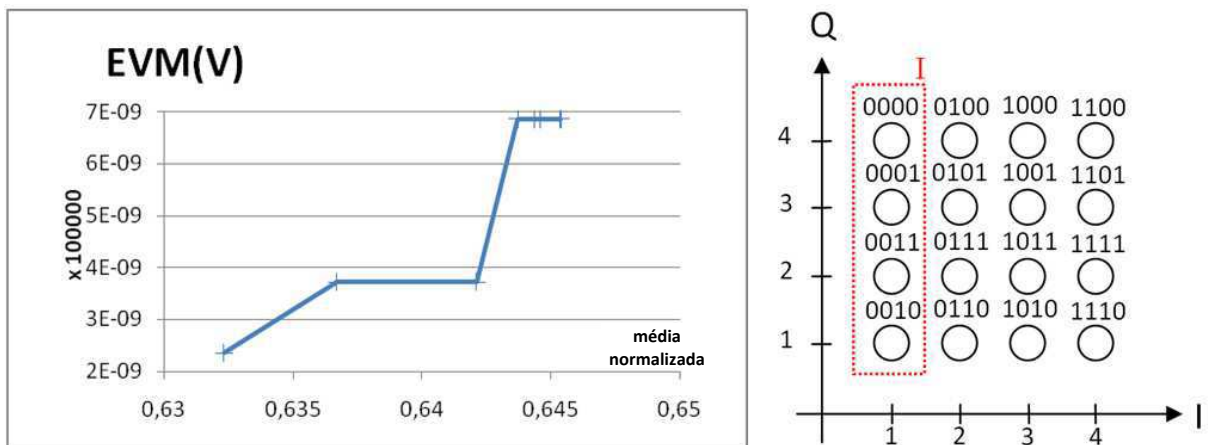


Figura 19 - Valores do EVM em relação à uma fraca cobertura dos parâmetros (entrada entre 0 e 3)

Mudando os estímulos de entrada para coberturas cada vez maiores, obtemos respostas de acordo com as Figuras 20, 21 e 22 para entradas **[0 ; 7]**, **[0 ; 11]** e **[0 ; 15]** respectivamente.

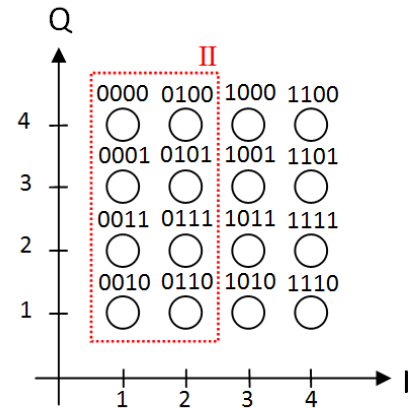
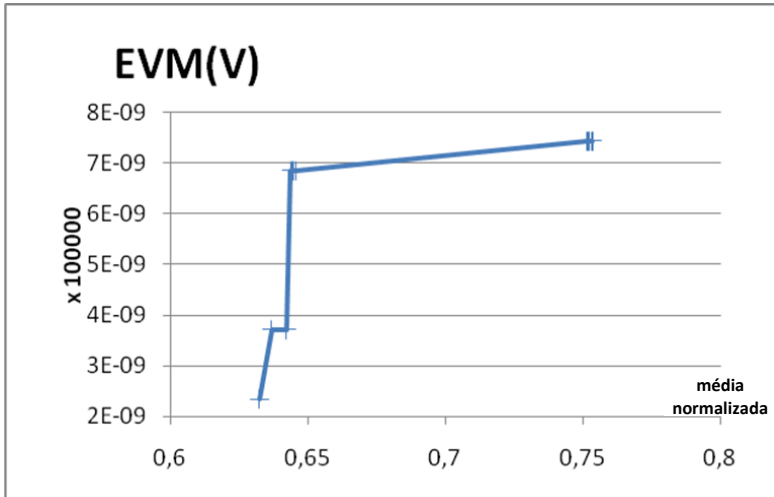


Figura 20 - Valores do EVM em relação à uma cobertura dos parâmetros (entrada entre 0 e 7)

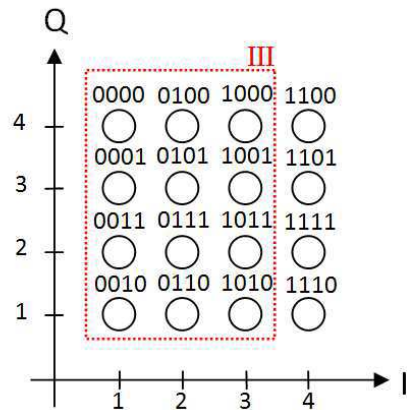
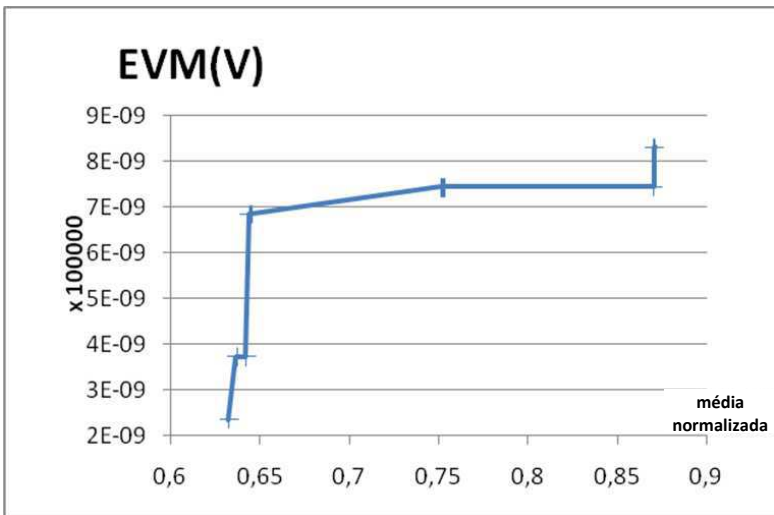


Figura 21 - Valores do EVM em relação à uma cobertura dos parâmetros (entrada entre 0 e 11)

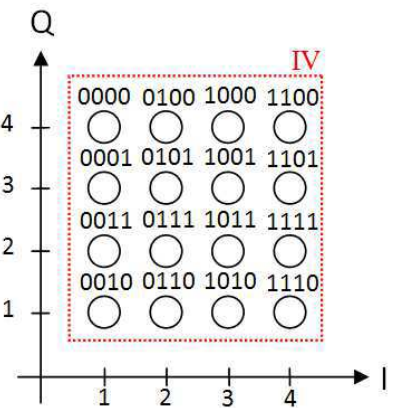
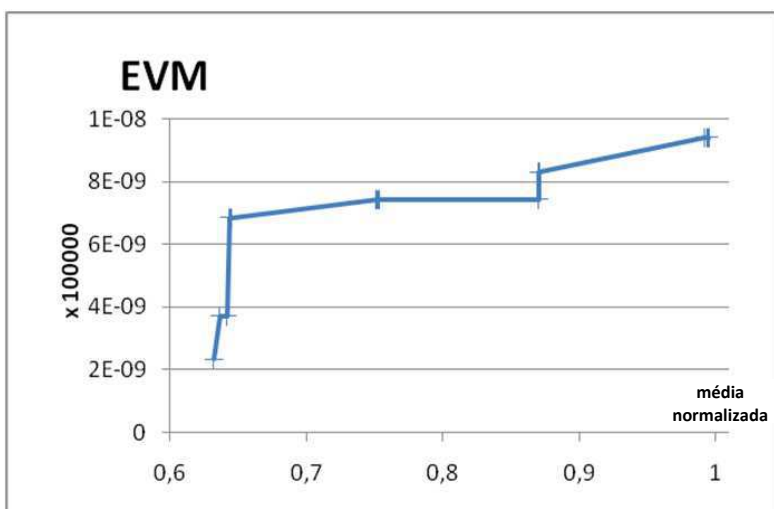


Figura 22 - Valores do EVM em relação à uma cobertura dos parâmetros (entrada entre 0 e 15)

Cada vez que aplicamos estímulos que possuem uma cobertura maior, percebemos que o *EVM* aumenta. Desta forma, conclui-se que existe uma correlação entre a cobertura dos parâmetros do sistema e o valor máximo do *EVM*. A Figura 23 mostra os quatro casos de estímulos aplicados.

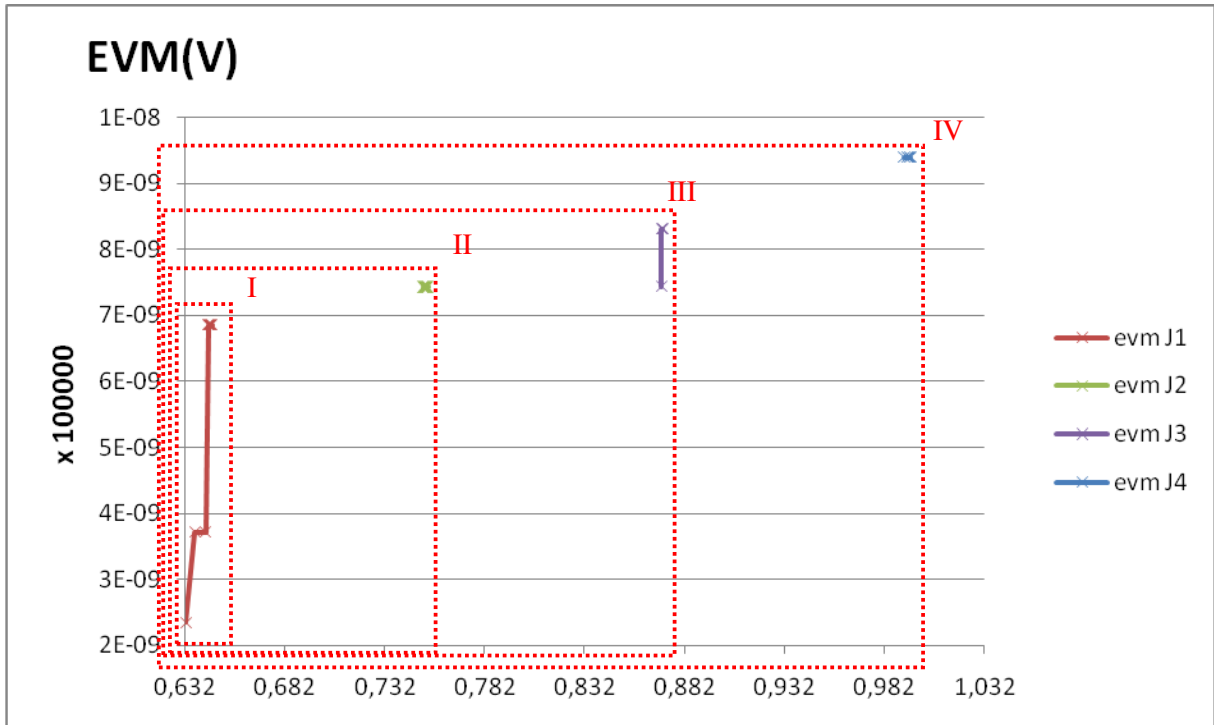


Figura 23 - Valores do *EVM* em relação à cobertura dos parâmetros (do mais fraco ao mais forte)

Analisando o gráfico da Figura 23, percebe-se que o *EVM* é constante na parte alta do gráfico, isto é, ele não evolui mais em relação à cobertura dos parâmetros do sistema.

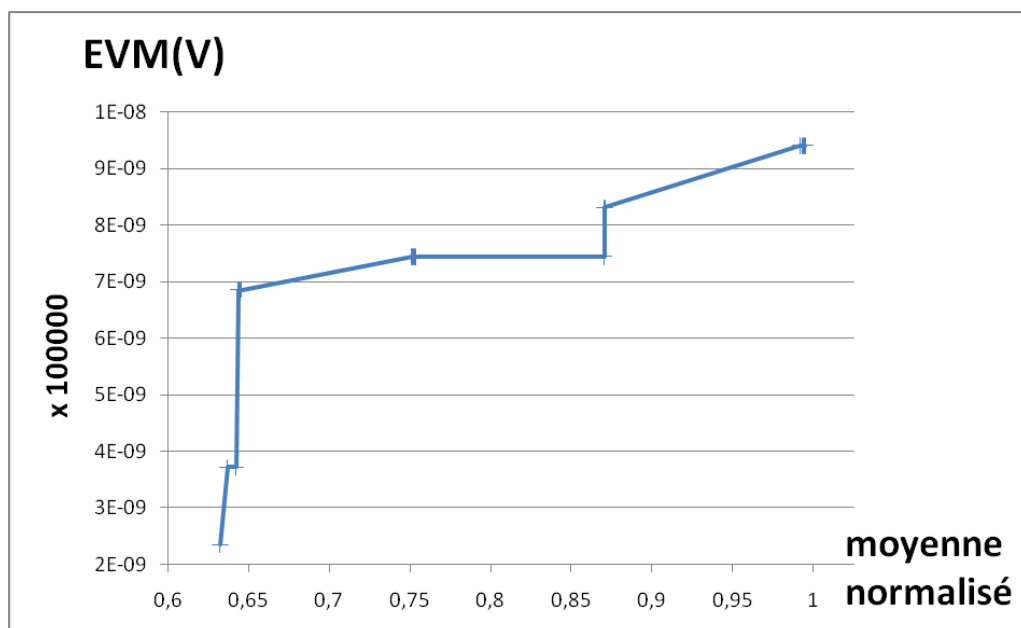


Figura 24 – Valores do *EVM* em relação à cobertura dos parâmetros do sistema

A escolha dos intervalos teóricos de variação de cada parâmetro do sistema é um fator muito importante para esta verificação. Ela serve de referência para cancelar a simulação quando o *EVM* não evolui mais. Se escolhermos os valores máximos teóricos do sistema, os valores próximos aos teóricos são esperados, mas, por exemplo, se escolhermos um intervalo de valores que é grande em relação à variação máxima, a cobertura será fraca.

O cálculo dos valores teóricos podem ser difíceis dependendo do sistema utilizado, mas neste caso, como se trata de um sistema simples, o cálculo dos intervalos dos filtros e dos outros componentes não é difícil. Se possuímos um sistema com vários amplificadores, filtros de ordens elevadas, componentes não-lineares etc, este cálculo se torna difícil.

4.4 Problemas de utilização do SystemC-AMS

SystemC-AMS trabalha com sinais analógicos e digitais. Os analógicos necessitam ser amostrados para que sejam tratados. Em relação à amostragem, nos deparamos com um grande problema quando trabalhamos com sinais analógicos de grandes frequências, pois a quantidade de amostras será muito grande. Para obter-se uma boa qualidade do sinal analógico, deve-se trabalhar em frequências de amostragem superiores à frequência de Nyquist³. Para que não haja interferência nos sinais amostrados, o critério de Nyquist deve ser obedecido.

Seja um sinal *M* de frequência máxima f_s com uma largura de banda *B*:

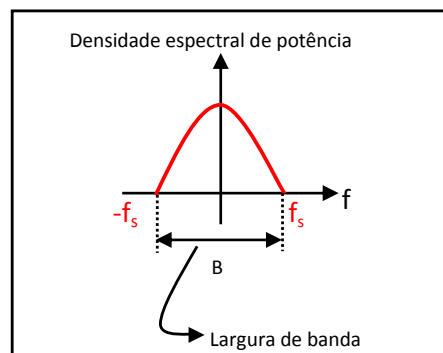


Figura 25 - Densidade espectral de potência

Se amostramos o sinal *M* com uma frequência f_e , uma translação do espectro do sinal centralizado nesta frequência é realizada. Se este valor for inferior ao dobro de f_s , nota-se que existem interferências (*aliasing*), mas quando a frequência de amostragem é superior ao dobro de f_s , não nos deparamos com este problema. A figura 26 mostra os três casos possíveis de se amostrar um sinal.

³ A frequência de Nyquist é igual ao dobro da frequência máxima do sinal

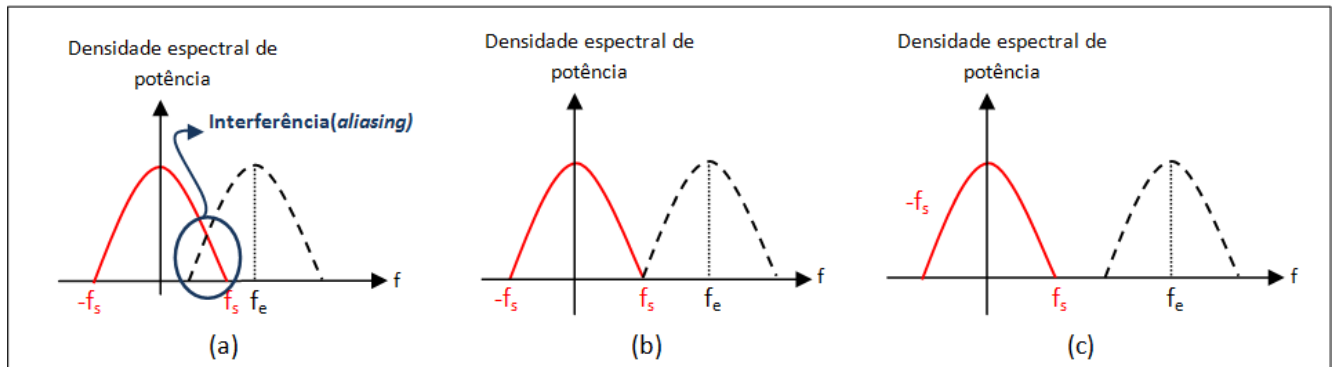


Figura 26 - Diferentes escolhas da frequência de amostragem: (a) $f_e < 2f_s$; (b) $f_e = 2f_s$; (c) $f_e > 2f_s$

Se amostrarmos o sinal com uma frequência de amostragem igual à de Nyquist, podemos representar o sinal, mas não com uma boa qualidade. A qualidade da representação do sinal é proporcional à f_e , mas deve-se escolher um valor que não seja tão grande, senão a quantidade de dados amostrados será muito grande.

O principal problema da verificação de sistemas mistos utilizando SystemC-AMS é quando trabalhamos com sinais de entrada de grandes frequências. Para simular sistemas RF que trabalham em altas frequências, cuja frequência da portadora é situada entre 3 e 30 MHz, necessita-se de um tempo de simulação longo e os arquivos gerados que contém os dados da simulação possuem demasiados dados para traçar as curvas.

5 Conclusão

Na análise da problemática que motivou o estudo deste estágio, nós conseguimos implementar um modulador QAM em SystemC-AMS, aplicando estímulos na entrada para realizar um estudo da evolução do EVM em relação à porcentagem da cobertura dos sinais internos do sistema.

SystemC-AMS é uma linguagem de descrição de hardware nova. Não existem muitos documentos sobre as funções que ele apresenta nas suas bibliotecas, e as utilizações de várias funções necessitaram ser analisadas diretamente através das linhas de código das mesmas.

Várias simulações foram realizadas, aplicando diferente estímulos de entrada para melhor cobrir os parâmetros do sistema para que o estudo da evolução do erro fosse feito. Os resultados das simulações confirmaram a correlação entre esta cobertura e o EVM. Este estudo permite um engenheiro de teste de circuitos integrados realizar testes mais precisos, aplicando estímulos que permitirão encontrar mais facilmente os erros presentes no sistema, diminuindo assim o seu tempo de validação.

6 Perspectivas

Os sistemas microeletrônicos sem fio estão cada vez mais presentes nas diversas soluções de engenharia. Devido a esta participação cada vez maior, é necessário que as linguagens de descrição de hardware realizem a simulação destes sistemas de uma forma mais eficiente para diminuir o tempo de produção dos mesmos.

Como já foi apresentado, SystemC-AMS possui uma grande dificuldade para trabalhar com sinais em altas frequências devido à quantidade de amostras que será feita, uma vez que a frequência da portadora é várias vezes superior à dos sinais de entrada.

Uma maneira de resolver este problema é de trabalhar no domínio da frequência. Para conseguirmos trabalhar neste domínio, é necessário aplicar uma transformada de Fourier em todos os sinais, fazendo que o tempo de simulação aumente e que o processo seja mais longo.

Atualmente, existe uma linguagem denominada SystemC-WMS (*SystemC Wave Mixed Signal*) que nos permite trabalhar com a representação de módulos através dos parâmetros a e b . [8] [9] [10] [11]

7 Referências Bibliográficas

- [1] <http://www.systemc.org> – último acesso em 9 junho 2009.
- [2] <http://www.systemc-ams.org> – último acesso em 9 junho 2009.
- [3] SystemC Analog & Mixed Signal. Extensions: What's It All About? – Martin Barnasconi, AMS WG Chairman – NASCUG IX, June 9, 2008.
- [4] VHDL-AMS and Verilog-AMS as Alternative Hardware Description Languages for Efficient Modeling of Multi-Discipline Systems – F. PÊCHEUX, C. LALLEMENT, Member IEEE, and A. VACHOUX, Member, IEEE.
- [5] SystemC: Methodologies and Applications, 273 – 297 – Chapter 10: SystemC-AMS: Rationales, State of Art, and Examples – Karsten EINWICH, Peter SCHWARZ, Christoph GRIMM and Christian MEISE.
- [6] Towards Analog and Mixed-Signal SoC Design with SystemC-AMS – Alain VACHOUX, Christoph GRIMM and Karsten EINWICH.
- [7] SystemC-AMS 0.15RC4 – Fraunhofer Institut Integrierte Schaltungen, February 2007.
- [8] SystemC-WMS: A Wave Mixed Signal Simulator – Simone ORCIONI, Giorgio BIAGETTI, and Massimo CONTI – Dipartimento di Elettronica, Intelligenza artificiale e Telecomunicazioni – Università Politecnica delle Marche, I-60131 Ancona, Italy.
- [9] Physical modeling in SystemC-WMS and real time synthesis of electric guitar effects – Francesco GAMBINI, Massimo CONTI, Simone ORCIONI, Franco RIPA, and Marco CALDARI - Dipartimento di Elettronica, Intelligenza artificiale e Telecomunicazioni – Università Politecnica delle Marche, I-60131 Ancona, Italy.
- [10] System Level Modelling of RF IC in SystemC-WMS – Simone ORCIONI, Mauro BALLICCHIA, Giorgio BIAGETTI, Rocco D. d'APARO, and Massimo CONTI - Dipartimento di Elettronica, Intelligenza artificiale e Telecomunicazioni – Università Politecnica delle Marche, 60131 Ancona, Italy
- [11] SystemC-WMS: Mixed-Signal Simulation Based on Wave Exchanges – Simoni ORCIONI, Giorgio BIAGETTI, and Massimo CONTI - Dipartimento di Elettronica, Intelligenza artificiale e Telecomunicazioni – Università Politecnica delle Marche, 60131 Ancona, Italy
- [12] <http://www.complextoreal.com/chapters/isi.pdf> - último acesso em 25 de maio de 2009.