

CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA



Universidade Federal
de Campina Grande

ROBERTO LUIZ PIMENTEL COSTA JUNIOR

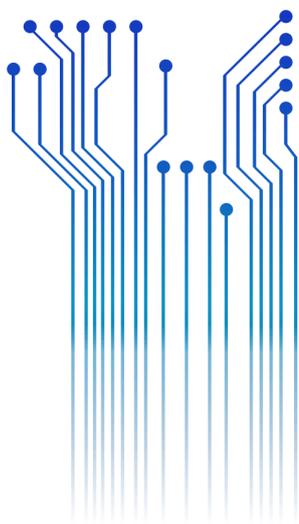


Centro de Engenharia
Elétrica e Informática

RELATÓRIO DE ESTÁGIO
NXP SEMICONDUCTORS



Departamento de
Engenharia Elétrica



Campina Grande
2018

ROBERTO LUIZ PIMENTEL COSTA JUNIOR

NXP SEMICONDUCTORS

*Relatório de Estágio Integrado submetido à
Unidade Acadêmica de Engenharia Elétrica da
Universidade Federal de Campina Grande
como parte dos requisitos necessários para a
obtenção do grau de Bacharel em Ciências no
Domínio da Engenharia Elétrica.*

Área de Concentração: Microeletrônica

Orientador:

Professor Gutemberg Gonçalves dos Santos Junior, D. Sc.

Campina Grande
2018

ROBERTO LUIZ PIMENTEL COSTA JUNIOR

NXP SEMICONDUCTORS

*Relatório de Estágio Integrado submetido à
Unidade Acadêmica de Engenharia Elétrica da
Universidade Federal de Campina Grande
como parte dos requisitos necessários para a
obtenção do grau de Bacharel em Ciências no
Domínio da Engenharia Elétrica.*

Área de Concentração: Microeletrônica

Aprovado em / /

Professor Gutemberg Gonçalves dos Santos Junior, D. Sc.
Universidade Federal de Campina Grande
Avaliador, UFCG

Professor Marcos Ricardo Alcântara Morais, D. Sc.
Universidade Federal de Campina Grande
Orientador, UFCG

Dedico este trabalho a todos aqueles que participaram de alguma forma desta jornada, sem vocês nada disso seria possível

AGRADECIMENTOS

Agradeço a primeiramente aos meus pais, por terem se esforçado para me proporcionar uma boa educação, e terem me alimentado com o apoio durante a minha estadia em Campina Grande, essencial para superar todas as adversidades ao longo dessa caminhada.

Agradeço também a minha namorada, Juliana de la Flor, por todo o tempo dedicado e apoio fornecido desde o início do curso.

Aos meus orientadores Prof. Marcos Morais, Prof. Elmar Melcher e Prof. Gutemberg Gonçalves, pelas oportunidades e orientações concedidas ao longo dos últimos anos.

A todos os membros do projeto de Excelência em Microeletrônica do Nordeste, por todos os ensinamentos que tive ao longo do tempo em que participei do laboratório, em especial Felipe Assis, por estar sempre disponível para tirar dúvidas quando o *StackOverFlow* não foi capaz de saná-las.

Enfim, agradeço a todos que de alguma forma, passaram pela minha vida e contribuíram para a construção da pessoa que sou hoje.

“A persistência é o menor caminho do êxito.”

Charles Chaplin.

RESUMO

Neste trabalho estão descritas as atividades realizadas durante o estágio realizado na equipe de Verificação de SoC do site da NXP Semicondutores no Brasil. Durante o período de estágio, 11 meses, foram realizadas atividades dentro da área de atuação da equipe, sendo produzidos estímulos para verificação dos componentes do projeto, apresentação e participação de seminários e desenvolvimento de *softwares* utilizados no contexto de verificação.

Palavras-chave: NXP Semicondutores, Verificação de SoC, Microeletrônica.

ABSTRACT

In this work is described all activities done during the internship in SoC Verification team on NXP Semiconductors' site in Brazil. During the internship, 11 months, activities were accomplished inside team's actuation area, been developed stimulus for verifying components of the project, presentation and participation of seminars and development of software used on verification flow.

Keywords: NXP Semiconductors, SoC Verification, Microeletronics.

LISTA DE ILUSTRAÇÕES

Figura 1 – NXP em Eindhoven, NL	13
Figura 2 – BSTC, Brazil Semiconductor Technology Center	14
Figura 3 – Fluxo do projeto	15
Figura 4 – Fluxo simplificado dos ambientes simulados.	20
Figura 5 – Exemplo de sinal propagando-se corretamente.....	22
Figura 6 – Exemplo de propagação de X após falha de setup de um sinal assíncrono ..	22
Figura 7 – Fluxograma de Resolução dos Tickets.....	24

LISTA DE ABREVIATURAS E SIGLAS

IP	Intellectual Property
RTL	Register-Transfer Level
SoC	System on Chip
VHDL	VHSIC Hardware Description Language

SUMÁRIO

Agradecimentos	v
Resumo	vii
Abstract.....	viii
Lista de Ilustrações	ix
Lista de Abreviaturas e Siglas	x
Sumário.....	xi
1 Introdução	12
1.1 Objetivo	12
2 Empresa	13
2.1 Atuação no Mercado	13
2.2 Brazil Semiconductor Technological Center	14
3 Fluxo de Projeto.....	15
3.1 Desenvolvimento dos IPs.....	15
3.2 Integração.....	16
3.3 Verificação.....	16
3.4 DFT.....	17
3.5 Testes	17
3.6 Validação	18
4 Atividades Realizadas	19
4.1 Treinamentos	19
4.2 Atividades Práticas	20
4.2.1 Verificação Funcional	21
4.2.2 Verificação Formal.....	22
4.2.3 Falhas na Verificação	23
4.2.4 Tickets	24
4.2.5 Documentação.....	25
5 Conclusão.....	26
6 Referências Bibliográficas	27

1 INTRODUÇÃO

Neste relatório de estágio busco descrever meu período como estagiário do setor de Verificação de Soc de 16/01/2018 a 17/12/2018 na empresa NXP Semiconductors. O estágio integrado é um componente curricular obrigatório, no curso de Engenharia Elétrica da Universidade Federal de Campina Grande – UFCG. E deve satisfazer uma carga horária mínima de 660 horas, seguindo os requisitos previstos na Resolução 01/2012 do Colegiado do Curso em consonância com a Lei do Estágio (Lei 11.788/2008).

1.1 OBJETIVO

Este trabalho tem como objetivo descrever as atividades realizadas durante o estágio na NXP, relacionando as experiências vividas no mercado de trabalho com os ensinamentos obtidos na Universidade. Será visto neste trabalho uma descrição sobre a empresa, seu histórico e atuação no mercado, além das atividades realizadas durante o período supracitado que se concentraram na área de Verificação de SoC, *System on Chip*, que trabalha fazendo a verificação à nível de sistema, simulando a interação do usuário final, através de *software*, com os periféricos conectados aos processadores.

A NXP é uma empresa com produtos de destaque na área de tecnologia, atuando fortemente nos mercados de microprocessadores e microcontroladores. Desta forma, a área de atuação da empresa se alinha com matérias da grade curricular, como Circuitos Lógicos, Arquitetura de Computadores e Arquiteturas Avançadas para Computação que explanam os mínimos conhecimentos necessários para trabalhar com os conceitos desenvolvidos dentro da empresa.

2 EMPRESA

A NXP Semiconductors é uma empresa de semicondutores com sede nos Países Baixos, fundada originamente em 1975 pela Philips, com nome Philips Semiconductors. Em agosto de 2006, a Philips anunciou a venda da sua divisão de semicondutores e a empresa passou a se chamar NXP Semiconductors. Atualmente, está presente em 33 países, possui 31.000 funcionários e cerca de 130 filiais pelo mundo (“About NXP | NXP”). Em dezembro de 2015, a NXP comprou a multinacional de semicondutores americana Freescale Semiconductors.

Figura 1 – NXP em Eindhoven, NL



Fonte: Dhanticounterfeit.com, 2018.

A principal missão da empresa é criar uma infraestrutura e conexões seguras para um mundo mais inteligente e criar soluções que tornem a vida das pessoas mais fáceis, melhores e seguras.

2.1 ATUAÇÃO NO MERCADO

A empresa tem várias linhas de produtos, dentre os quais pode-se destacar os chips e microcontroladores para as áreas automotiva, IoT, segurança, saúde e telefonia. As principais indústrias que desenvolvem e fabricam produtos nestes segmentos no Brasil

são clientes da NXP. Ainda, por tratar-se de uma multinacional, detém clientes de grande renome, como Amazon, Apple, Bosch, Continental, Ericsson, Facebook, Gemalto, Giesecke & Devrient, Huawei, Hyundai, Kona, Nokia Networks, Panasonic, Samsung e ZTE.

2.2 BRAZIL SEMICONDUCTOR TECHNOLOGICAL CENTER

O estágio foi realizado no centro de pesquisa da NXP localizado em Campinas – SP e conhecido como Brazil Semiconductor Technological Center – BSTC. A equipe de funcionários é dividida em subgrupos, entre eles, os times de verificação e integração de SoC, verificação e design de *IP*, validação e desenvolvimento analógico.

Figura 2 – BSTC, Brazil Semiconductor Technology Center



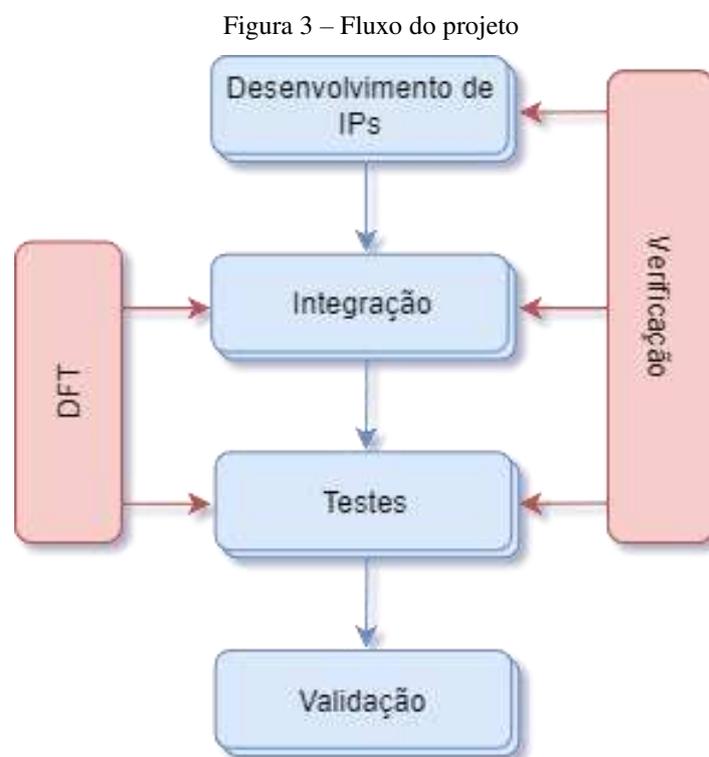
Fonte: Google, 2018.

O BSTC iniciou as operações em 1997 quando ainda atuava sob a direção da Motorola Inc. que apostou na capacidade intelectual e de inovação de engenheiros locais. Apesar de ter começado com apenas 8 engenheiros, o centro conta atualmente com 140 funcionários, dos quais 95% são engenheiros (“NXP no Brasil | NXP”).

Durante os seus 21 anos de história, o BSTC produziu mais de 100 projetos, incluindo microcontroladores e IPs em áreas como gerenciamento de potência, redes automotivas, processamento digital de sinais e aceleradores criptográficos e de temporização.

3 FLUXO DE PROJETO

O processo de criação de um chip é longo e envolve uma grande quantidade de profissionais, cada um especializado numa diferente etapa do processo. Dentro deste, vários grupos contendo etapas acabam sendo classificados sob identificadores, que auxiliam os gerentes e líderes a identificar o estado atual do projeto e poder estimar as ações necessárias para otimizar o processo.



Fonte: O próprio autor.

3.1 DESENVOLVIMENTO DOS IPS

A primeira etapa no fluxo envolve a criação dos IPs, ou *Intellectual Properties*, componentes que possuem funcionalidades específicas desejáveis no chip. Esta etapa envolve a criação tanto de IPs digitais quanto analógicos, podendo existir ainda, os IPs que possuem ambos funcionando em sincronia.

Os IPs digitais são sintetizáveis através de ferramentas de linguagem descritivas, o que torna sua construção possível através de estruturas lógicas básicas, como portas AND, OR, XOR, NAND, NOT e NOR.

Os IPs analógicos são as estruturas que têm funcionalidade específicas que ferramentas são incapazes de gerar de forma digital. Desta forma, essa etapa requer o desenvolvimento a nível de transistores e estudos dos vários limites eletrônicos para o componente, afim de que atendam o necessário para a aplicação ao qual estão sendo desenvolvidos. Em alguns casos de IPs muito complexos, é necessário a criação de uma interface de comunicação digital para controlar a parte analógica. Desta forma, é desenvolvido um controlador digital que irá facilitar a troca de informações entre o IP analógico e o resto do SoC.

3.2 INTEGRAÇÃO

Uma vez criada ou reutilizada a lista dos componentes específicos necessários, é possível prosseguir para a etapa de integração. Nesta, os IPs são conectados entre si e suas funcionandas são aplicadas de forma que as especificações do *SoC* sejam atendidas. Dependendo dos IPs conectados, e da maneira que o são, é possível gerar funcionalidades mais complexas e resultar em um produto com características finais diferentes, como potência, consumo ou leakage. Paralelamente a essa conexão, é necessário distribuir os componentes no espaço físico destinado ao chip, tomando cuidado com as posições de cada IP para que as conexões não fiquem muito espaçadas ao ponto de gerar problemas de sincronismo no sistema.

É ao final da etapa de integração que o projeto sai do ambiente de simulações e passa para a etapa real, com produção em silício contendo o chip desenvolvido. A partir deste ponto, é possível fazer o controle de qualidade e realizar as análises que vão de fato caracterizar o comportamento do produto real, e não mais simulado.

3.3 VERIFICAÇÃO

Paralelamente a todo esse fluxo de desenvolvimento do produto, é necessário garantir que não hajam erros ao longo do projeto. Quaisquer erros que possam se propagar sem controle estariam indesejadamente presentes no resultado final. Desta maneira,

existem divisões responsáveis por verificar cada uma das etapas mencionadas anteriormente.

De maneira mais abstrata, é possível dividir essas divisões em dois grandes grupos: o de verificação de IPs e o de verificação de SoC. Tendo o foco de suas atenções em detectar erros nas funcionalidades e na integração dos IPs, respectivamente. Dessa maneira, o grupo de verificação de um projeto começa seu trabalho já no início do fluxo e continua trabalhando ativamente até o momento em que não houver mais nenhuma alteração física ou lógica no produto.

3.4 DFT

A etapa de DFT ou *Design For Testability*, é responsável por adicionar, principalmente durante a integração, componentes e lógica no SoC que permitam que o time de testes efetue, nas etapas futuras, testes que proporcionem ao projeto a maior taxa de cobertura possível. Ao testar o chip, é possível detectar discrepâncias entre a lógica simulada e o resultado físico, promovendo assim, maior segurança e qualidade para o projeto.

Esta etapa, é considerada parte da Integração, devido às atividades desenvolvidas e ao período de atuação no fluxo do projeto. Sua representação única no fluxo se faz necessária, entretanto, por sua atuação persistir mesmo após o tapeout, continuando trabalhando conjuntamente com a equipe de testes. Nessa fase, geralmente existe a geração de novos padrões de cobertura para oferecer melhores resultados e análise de eventuais correções necessárias ao RTL. Uma testabilidade eficiente garante que problemas de fabricação sejam detectados e que as peças defeituosas sejam descartadas antes de ser enviada ao cliente.

3.5 TESTES

Já com o chip fisicamente disponível, a etapa de testes consiste em verificar que os padrões e rotinas, geradas na integração, retornem os resultados esperados e definidos em simulação quando exercitados. Esses testes são realizados através de máquinas especialmente desenvolvidas, chamadas de testadores. Elas possuem agulhas ou soquetes desenvolvidos especificamente para cada projeto, e são programadas para gerar valores

iguais aos padrões previamente definidos, e comparam com os resultados esperados, testando assim, a funcionalidade da parte lógica do chip e verificando se houve algum erro de fabricação no chip.

3.6 VALIDAÇÃO

Por fim, ocorre a etapa de validação, onde diversas análises realizadas no chip confirmam suas características e especificações determinadas no início do projeto. Realizando ainda testes com determinadas condições de funcionamento que simulam o envelhecimento do chip, o que permite especificar como será o seu comportamento ao longo dos anos.

4 ATIVIDADES REALIZADAS

O estágio foi realizado na equipe de Verificação de *SoC*, que é responsável pelo processo de verificação da conexão entre os diferentes IPs utilizados nos *SoCs* da NXP. O objetivo da equipe é não só garantir a integridade das conexões feitas entre os componentes utilizados, como também de garantir que a experiência do usuário final estará de acordo com as especificações do projeto. Isso inclui então, verificar o funcionamento dos processadores e aceleradores, escritas e leituras na memória, condições de consumo, e principalmente se as interligações de cada *IP* estão sendo realizadas. Portanto, é de responsabilidade da equipe elaborar um plano de verificação e garantir que todas as possíveis falhas de integração possam ser corrigidas através desse plano.

Diferentemente da equipe de verificação de *IP*, o foco dessa verificação não é em garantir que cada *IP* está cumprindo com sua funcionalidade, portanto, alguns objetivos compartilhados têm menor prioridade. Dando-se importância maior para a garantia de que todas as entradas e saídas estão sendo estimuladas corretamente, do que se todos os modos de operação de cada *IP* estão funcionando corretamente. Partindo deste princípio, as atividades práticas desenvolvidas durante o período de estágio giraram em torno de criações de teste em C que estimulassem todas as conexões dos *IPs* a serem verificados, e pudessem simular o uso normal por parte do usuário final.

Além dessas atividades práticas, foram oferecidos alguns treinamentos no início do estágio, principalmente sobre ferramentas da própria empresa, embora tenham sido ofertados também cursos para maior familiarização com as linguagens básicas para a realização das atividades práticas.

4.1 TREINAMENTOS

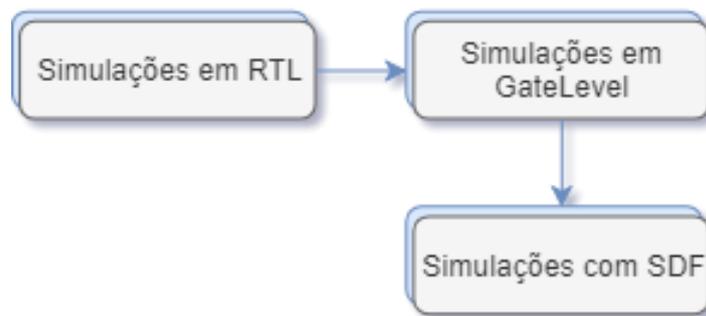
No início do processo, foi necessário um estudo aprofundado e treinamento com os funcionários mais experientes, a fim de que todos os estagiários ficassem a par de diferentes ferramentas, sistemas e protocolos utilizados na empresa, para que todos conhecessem as diferentes áreas e etapas do processo de desenvolvimento ao qual

estariam diretamente envolvidos. Inicialmente, os treinamentos foram focados nos conceitos de qualidade e metodologia aplicados na empresa, mas ainda tiveram abordados diversos outros, focados nos fluxos de projeto, sistemas de segurança, utilização e funcionamento de um sistema de controle de versão específico da empresa, linguagens descritivas e desenvolvimento de algoritmos.

4.2 ATIVIDADES PRÁTICAS

Simplificadamente é possível resumir a atuação da equipe de verificação de *SoC*, em três grandes ambientes de trabalho, onde o que muda de um ambiente para o outro, é que está sendo simulado: o RTL, o Netlist – que nada mais é do que o RTL compilado, descrito através de portas lógicas – e o Netlist com os atrasos dos caminhos adicionados, ou Netlist com *SDF*, *Standard Delay Format*.

Figura 4 – Fluxo simplificado dos ambientes simulados.



Fonte: O próprio autor.

Como foi possível ao estagiário ingressar em um projeto desde a concepção do plano de verificação, foi possível a sua atuação em todas essas etapas. Desta forma, durante o período de estágio, foi possível a participação em diversas frentes de conhecimento, sendo as mais importantes:

- Desenvolvimentos de *software* para verificação funcional e formal;
- Estudo de ferramentas para detecção de falhas na verificação;
- Abertura de tickets para que sejam corrigidas as falhas encontradas;
- Documentação das atividades desenvolvidas;

Estas atividades serão explanadas de forma mais detalhada nos tópicos a seguir.

4.2.1 VERIFICAÇÃO FUNCIONAL

De forma geral, todo o fluxo da verificação de SoC gira em torno desta atividade, a verificação de que os *IPs* integrados pela equipe de *Design*, estão funcionando de acordo com os especificados na documentação. Então, é tarefa da equipe, desenvolver o *software* que será executado pelos processadores, ou aceleradores disponíveis no *SoC* e estimular cada *IP*. Para atingir tal objetivo, configura-se as diversas opções de *clocks*, *resets*, *triggers externos* e demais opções do *IP* de forma a garantir que todos os sinais que entram e saem dele, estão corretamente conectados.

Dado um projeto em execução, os *IPs* e aceleradores especificados são distribuídos entre os membros da equipe de Verificação de *SoC*, sendo o estagiário responsável pela verificação da *GPIO*, do controlador de *IO*, de alguns temporizadores, algumas funcionalidades do *DSP* e verificação de estímulos de *stress* no *SoC*. Desta forma, detectou falhas na integração dos pinos de entrada e saída do *SoC* e problemas nas configurações dos estados de baixo consumo de energia em que o time de *Design* foi prontamente notificado para que fossem corrigidos.

Como uma das filosofias que norteiam as equipes de verificação e integração é sempre a reusabilidade do trabalho, uma boa parte do trabalho nesta parte consistiu apenas em adaptar os *softwares* existentes dos *IPs* para o *SoC* em questão. No entanto, em alguns casos existe uma nova funcionalidade no projeto em que é necessário o desenvolvimento de um novo estímulo que teste essa particularidade.

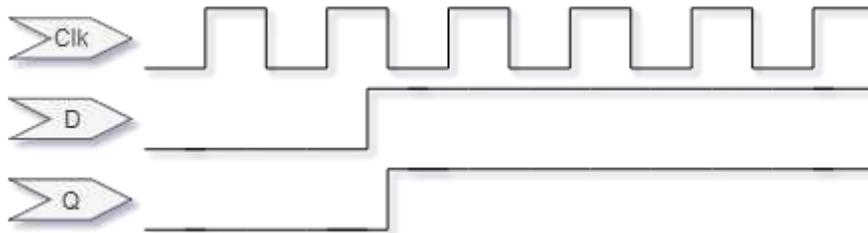
Finalmente, após algumas versões do *RTL* disponibilizadas pelo time de Integração é proposta uma como a Versão Final. Onde após o lançamento desta, ocorre a etapa de síntese do *RTL*, e então, a disponibilização de uma *Netlist*.

Com este *Netlist*, a equipe de verificação garante que todas as funcionalidades que estavam descritas no *RTL* foram de fato transferidas para a síntese, e estarão presentes no silício. Para isto, é rodada uma nova série de testes funcionais direcionados para as áreas mais sensíveis de erros nesta etapa, como quando ocorrem modificações nos estados de potência do *SoC* ou quando um sinal atravessa domínios de *clock*.

Idealmente, após esta etapa, os sinais devem se propagar com os atrasos obedecendo as especificações de frequência estipuladas. No entanto, é sabido que nem todos os sinais do *SoC* irão se comportar desta maneira, pois existem diversos fatores que influenciam nesse comportamento ideal. Daí uma importante tarefa da equipe de verificação nessas simulações é a de mapear os registradores que irão eventualmente

desrespeitar as condições de setup-hold, mas onde este já é um comportamento esperado, e estes registradores são utilizados de forma a garantir a correta propagação dos sinais em condições assíncronas, sendo chamados sincronizadores.

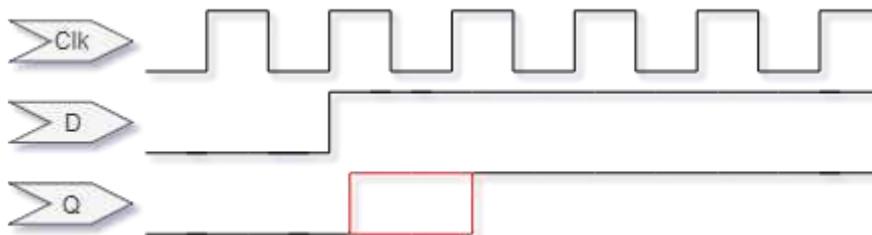
Figura 5 – Exemplo de sinal propagando-se corretamente.



Fonte: O próprio autor.

Após serem mapeados os sincronizadores, os resultados de simulação com atrasos devem ser os mesmos das simulações em RTL e da simulação sem atrasos.

Figura 6 – Exemplo de propagação de X após falha de setup de um sinal assíncrono



Fonte: O próprio autor.

É também nesta etapa de pós-síntese que são realizados os testes necessários para simular que o *SoC* é capaz de manter as suas especificações de potência e frequências nas melhores e piores condições de temperatura e tensão.

4.2.2 VERIFICAÇÃO FORMAL

Apesar da verificação funcional ser planejada e desenvolvida para cobrir funcionalmente todas as possíveis fontes de erros, alguns sinais não podem ser corretamente cobertos por este modelo de verificação por motivos de limitação da ferramenta, como por exemplo, configurações dos pinos externos que aumentam ou diminuem a corrente nos transistores, realizando assim um controle na velocidade com o qual o sinal se propaga para dentro ou para fora do *SoC*. Apesar de ser uma funcionalidade dos pinos, não pode ser emulada pela ferramenta de verificação, resultando em um sinal

exatamente igual do ponto de vista do simulador e que, portanto, não pode ser verificada de formas tradicionais. Para este caso, e alguns outros, é utilizada a verificação formal.

A verificação formal nada mais é do que a definição de axiomas dentro do contexto de verificação, sendo comumente utilizada para a verificação de protocolos de comunicação, onde as partes determinam uma sequência de sinais que devem ser obrigatoriamente obedecidas, com padrões definidos para casos de falha e sucesso da transação.

Foi utilizada para verificação da propagação de sinais elétricos que não podiam ser estimulados funcionalmente, e também para a verificação de alguns Mux's presentes no *RTL*, onde o estímulo destes por meio da verificação funcional é altamente custoso, podendo ser facilmente modelado na verificação formal, além de trazer uma confiança a mais no âmbito geral da verificação, aumentando a confiabilidade do *RTL* simulado.

4.2.3 FALHAS NA VERIFICAÇÃO

Por fim, uma importante atividade desempenhada durante o período de estágio, foi o estudo de uma ferramenta capaz de detectar falhas nas etapas de verificação do SoC.

A ferramenta insere de forma simples, em uma análise dos padrões de verificação formal e funcional, verificando quais padrões estimulam cada sinal de entrada e saída nos blocos simulados e através desses dados, insere numa cópia do RTL falhas de design.

Teoricamente, como os padrões estimulam os locais onde os quais foram inseridas as falhas de design, estes padrões devem ser capazes de detectar essas falhas e acusar os erros inseridos. Na prática, o que foi visualizado, é que os padrões verificados, apesar de cobrirem todos os sinais dos protocolos de comunicação, ainda apresentavam algumas falhas, o que fez, mais de uma vez, com que os erros inseridos no RTL pela ferramenta não fossem detectados pelos padrões. Então, a partir destes resultados, alguns testes foram reescritos, e outros reestruturados, de forma que fosse possível cobrir todos os erros inseridos pela ferramenta.

Apesar dos ótimos resultados exibidos pela ferramenta, foi levado em consideração a demora da mesma para que fossem observados os seus resultados. Casos onde normalmente uma regressão de um único bloco pequeno demorava algo entre 30 minutos e 2 horas para apresentar os relatórios de cobertura. Quando acrescenta-se o tempo de compilação e execução até chegar ao ponto dos relatórios da ferramenta com o

mesmo bloco demorar de 6 até 12 horas, onde o caso de um bloco maior chegou a demorar 35 horas.

Finalmente, foi proposto que a ferramenta fosse incorporada ao fluxo de verificação de SoC, mas não como foco inicial da equipe, e sim como um relatório final, à ser aplicado após o fluxo padrão de verificação, como uma forma de que a equipe tivesse um feedback se os testes escritos realmente seriam capazes de detectar os erros inseridos no RTL.

4.2.4 TICKETS

Durante o fluxo de verificação é normal a detecção de diversos problemas nas versões do design liberadas pelo time de integração, seja nas conexões entre os IPs ou em componentes específicos. Desta maneira, uma importante atividade era a de comunicar ao time responsável os bugs encontrados e se possível, a forma de resolução dos mesmos.

Figura 7 – Fluxograma de Resolução dos Tickets.



Fonte: O próprio autor.

A metodologia para resolução de bugs, conforme pode ser observado de forma simplificada na Figura 7 – em que as etapas destacadas em vermelho são desempenhadas pelo time de desenvolvimento, seja ele de IP ou de integração, e as etapas em azul eram conduzidas pelo time de verificação – se dava da seguinte maneira: Liberada uma nova versão do RTL ou do Netlist, toda a equipe de verificação deve migrar para esta versão e começar o processo de depuração dos erros. Uma vez identificado a causa raiz do mesmo, é aberto um chamado para o time responsável pelo mesmo, podendo ser analisado pelo

desenvolvedor do IP ou pela equipe de integração, dependendo de onde a falha foi encontrada. Uma vez aberto o chamado, a equipe de desenvolvimento analisa se é realmente uma falha, que uma vez confirmada exigia correção necessária por parte do time de desenvolvimento, sendo normalmente liberada na versão seguinte do RTL ou Netlist. Em alguns casos, falhas mais graves eram encontradas que impediam o processo de verificação até que fosse corrigida. Nestes casos, uma correção temporária era adicionada ao testbench de verificação e liberada assim que confirmada a existência da falha, de forma a permitir que o fluxo de verificação não fosse atingido como um todo.

Em todos os casos, a versão seguinte liberada pelos times de desenvolvimento geralmente continha a solução do problema detectado, embora fosse possível que as diversas interpretações possíveis da especificação do projeto escalassem o problema para o arquiteto de sistemas, que deveria finalmente explicar melhor o requisito desejado e sanar todas as eventuais dúvidas de todos os envolvidos.

4.2.5 DOCUMENTAÇÃO

Como o fluxo do projeto envolve diversas pessoas, entre elas pessoas de outros locais de atuação da NXP e até pessoas de outras empresas, uma tarefa que é bastante disseminada, consiste no compartilhamento do conhecimento entre as equipes. Desta forma, o estagiário participou de diversas reuniões com times de Verificação de IP, Desenvolvimento de Memórias, Verificação de SoC, além de palestrantes externos de forma que o conhecimento pudesse ser compartilhado entre todos os presentes.

Então, diversas vezes, quando uma pessoa estava envolvida em uma atividade muito específica para resolução do problema de verificação, ou era adotada uma estratégia diferente da tradicional aplicada ao fluxo, esta pessoa apresentava um rápido seminário com o objetivo de que todos pudessem ter um ponto de partida quando encontrassem problemas semelhantes.

A utilização da verificação formal como alternativa de verificação também resultou em um pequeno seminário para os líderes de projeto por parte do estagiário. Ao final, foi escrito um guia explicando como executar as ferramentas utilizadas de forma a gerar o mesmo resultado.

5 CONCLUSÃO

As atividades designadas ao estagiário foram cumpridas com êxito, estando o estagiário por mais de uma vez envolvido em linhas de atuação consideradas fundamentais para o projeto.

O ambiente da NXP mostrou-se um excelente local para o aprendizado do fluxo de projeto em microeletrônica, estando de acordo com disciplinas do fluxograma do curso de Engenharia Elétrica.

Ao final do período de estágio, foi possível observar que devido à realização de atividades muito específicas do setor de atuação, o período de um ano realizando o mesmo é extremamente necessário para a completa inserção do estagiário no fluxo da equipe de verificação, sendo que um período de pelo menos 4 meses fora dedicado exclusivamente para a adaptação do estagiário.

Desta forma, é importantíssimo frisar que a adaptação do currículo do curso de forma a permitir estágios mais longos é fundamental para a inserção do aluno da UFCG neste meio.

6 REFERÊNCIAS BIBLIOGRÁFICAS

About NXP. Disponível em: <<https://www.nxp.com/about/about-nxp/about-nxp:ABOUT-NXP>>. Acesso em: 4 dez. 2018.

NXP® no Brasil. Disponível em: <<https://www.nxp.com/about/about-nxp/about-nxp/worldwide-locations/nxp-no-brasil:BRAZIL>>. Acesso em: 4 dez. 2018.

MANO, M. M.; CILETTI, M. D.; MANO, M. M. **Digital design.** Pearson Prentice-Hall, 2007.

MANO, M. M.; KIME, C. R. **Logic and computer design fundamentals.** Pearson/Prentice Hall, 2006.