

Phases

segunda-feira, 6 de setembro de 2021 14:46

[02 - Digital 01](#) ([Web view](#))

- **Macro phases: functional, electrical and geometrical**
 - Três pontos de vista relacionados com as entregas: funcional (incipiente), elétrico (muito informal) e geométrico (bem avançado com espaço de melhora)
 - como floorplan e posicionamento de pinos (LEF/DEF), qual geometria usar.
 - Diferentemente, do ponto de vista funcional e elétrico (meio termo entre parte funcional e geométrica, relacionando tempo de transição, capacitância).
- **Geometrical phase more mature than others**
 - Eu vejo a troca entre times acontecendo no floorplan/pinos e depois na entrega, quando você integra a biblioteca OA lib, passar DRC/LVS e envia pro analog integrar.

Geometrical

segunda-feira, 6 de setembro de 2021 14:48

[02 - Digital 01 \(Web view\)](#)

- **Related to backend, layout**
 - não mexo muito com o backend, mas participei de coisas interessantes, como floorplan e posicionamento de pinos (LEF/DEF), qual geometria usar.
 - Eu vejo a troca entre times acontecendo no floorplan/pinos e depois na entrega, quando você integra a biblioteca OA lib, passar DRC/LVS e envia pro analog integrar.
- **Main and most mature interaction point between teams currently**
 - não mexo muito com o backend, mas participei de coisas interessantes, como floorplan e posicionamento de pinos (LEF/DEF), qual geometria usar.
 - avançamos significativamente nesse ponto, onde temos a definição dessa geometria desde o início do projeto e refinado ao longo do projeto.
 - Também entra as ferramentas de versionamento para deixar mais transparente,
 - Essa troca aconteceu várias vezes, pessoal entendeu e aceitou, foi rápida e tranquila.
 - "Temos um caminho claro e a aderência tá boa"
 - Eu vejo a troca entre times acontecendo no floorplan/pinos e depois na entrega, quando você integra a biblioteca OA lib, passar DRC/LVS e envia pro analog integrar
 - com todos falando a mesma língua de DRC e LVS,
- **Chipus differential advantage**
 - Está melhorando e ainda pode melhorar, onde traz um grande diferencial para a empresa pois trabalhos com muitos serviços independente da tecnologia.
- **Bigfiles problem**
 - o problema dos bigfiles... é um ponto aberto que ninguém pegou essa bola ainda. É uma questão global que vai acontecer e que está sendo negligenciado porque não deu problema ainda.

Electrical

segunda-feira, 6 de setembro de 2021 14:50

02 - Digital 01 ([Web view](#))

- **Related to transition times and load**
 - elétrico (meio termo entre parte funcional e geométrica, relacionando tempo de transição, capacitância)
 - Por regra (99.99% dos casos), nós não temos requisitos elétricos, um handshake, os DRVs e etc...
- **Current challenge**
 - "toda a parte de frontend (funcional e elétrica) são os gargalos".
- **Missing convergence and clarity**
 - A parte elétrica falta uma convergência (um documento, um entendimento... um pouco dos dois talvez)... é como se o digital tivesse um placeholder mas como fazer essa informação trafega de uma forma lisa assim como a parte geométrica (onde pegar, onde commitar, etc).
 - Essa parte na visão elétrica não tá clara ... cara consiga ter isso como requisito do seu bloco.
 - a questão da interface não tá claro como requisito do bloco digital,
 - As vezes é uma informação que já tem (qual a carga default para fazer a simulação?) e que precisa ser passado para o outro lado de quem não tá projetando o bloco. Não está claro qual informação o outro lado precisa, o que facilitaria para identificar pontos faltantes.

Functional

segunda-feira, 6 de setembro de 2021 14:50

02 - Digital 01 ([Web view](#))

- **Related to behaviour of circuit**
 - A parte da simulação AMS
- **Current challenge**
 - "toda a parte de frontend (funcional e elétrica) são os gargalos".
- **Essentially a spec problem**
 - Do ponto de vista funcional, temos o problema de como é que se chega de uma spec para um circuito digital? "De onde vem a spec"?
 - em uma questão cultural e como viabilizar uma estratégia top-down. Tá relacionado a especificação sistêmica / alto nível.
- **Missing team interactions**
 - A parte da simulação AMS está tentando abrir essa caixa preta em ambos times, mas precisa ter um conhecimento em comum de simuladores, metodologias, opções... isso tudo tem que ficar mais claro, faltando transparência e conversa. Faltou essa parte de interagir mais, trocar mais figurinha. Isso deve refletir sendo uma preocupação desde o início no projeto, não podendo ser deixado pro final pois sempre tem causado problemas e não fica tão bom quanto poderia.

Discipline visions

segunda-feira, 6 de setembro de 2021 14:53

Renamed from "clarity" to "mixed-signal vision" because was always used in this context by Digital 01. Then merged with "Discipline visions"

01 - Manager 01 (Web view)

- **Share vision both ways to specify importance**
 - A troca de entregas sempre trás **abordagens diferentes** por causa dessa **visões relativas de cada disciplina (analog e digital)**.
 - floorplan (abstract view em formato LEF). Nessa parte, eu acho que é preciso saber **o que vai ser feito do outro lado** e **formalizar** as definições.
 - "OK, o digital vai ser esse quadrado", mas são **definições muito incipientes**: quais as posições dos pinos? Quais os nomes dos pinos?
 - Eu tenho noção no que as entregas são usadas pela outra disciplina, mas **não acho que isso está claro** para todos os envolvidos.
 - No mixed-signal, mostra que não basta ter uma visão clara do sistema do ponto de vista do analogico ou do digital, da **visão de uma disciplina**, mas uma **visão mixed**.
 - ... problemas que ficaram meio que na **zona cinza**, como uma feature que é controlada pelo digital, mas o entendimento da feature é mixed pois estava relacionado com medição de corrente. Por isso, na hora de **especificar e validar as features**, tem que ter uma **visão mixed no topo** e as duas **disciplinas tem que se entender** (as vezes até **considerar extra-chip**: dependendo dos instrumentos de medição que serão usados ou testes em silicio) . Essa feature foi pra **spec pouco detalhada** e, apesar de muito simples, tinha de ser visto na **perspectiva do sistema**.
- **Not considering about the other discipline is a source of problems**
 - blocos de outra disciplina geralmente são considerados como **caixas pretas durante muito tempo** ...
 - Ainda no LIB, se o projetista analog **não tem essa clareza ou a ferramenta** (scripts) para isso, o projetista digital **precisa gerar (ou pedir) o que ele precisa usar** ao inves de apenas receber.
 - **Para refinar** o que tem de ser feito, geralmente tem **muitas interações**. Como geralmente ocorrem **muito tarde**, acaba tendo que **sacrificar alguma coisa** (qualidade), como checagens ou não cobrir algo.
 - ... ECO novamente por causa de **problema com pinos** (pinout errado em arquivos). ... é um **tipico ponto** que **não é uma preocupação pro projetista analog** ("**botei qualquer direção**"), mas resultava num curto quando a lógica estivesse em alto... um pino estava com uma **propriedade errada**.
 - Interface é a parte onde chega o que você não sabe ("**eu conheço até aqui, tô trabalhando nessa sala**"), e a mesma coisa do outro lado ("**na outra sala**"). Esse problemas mostram que "**logo ali tá ficando um ponto cego**". "**Os dois pararam de enxergar um pouco antes e ficou um buraco**".
- **Independent top verification**
 - Quando chega na **verificação de topo**, informações de como ou quais testes serão feitos fica **centralizado no responsavel** pela integração ou topo com **pouco envolvimento de outros**.

02 - Digital 01 (Web view)

- **Share vision both ways to specify importance**
 - Acho que é mais o ponto de **especificação, onde e como colocar essa spec**. Tem de ser **simples para ter uma clareza**.
 - **Não está claro qual informação o outro lado precisa**, o que facilitaria para **identificar pontos faltantes**. Os times deveriam fazer uma **conversa para ficar bem claro esse processo e conseguir ver o que tá sendo usado, em que documento isso tá registrado e assim vamos convergir**. "**O melhor não é perguntar para o projetista, é ter um acordo de onde fica registrado**".
 - Tá relacionado a **especificação sistêmica / alto nível**. A parte da **simulação AMS** ... tem que ficar mais claro, faltando **transparencia e conversa** (entre os times). Faltou essa parte de

- interagir mais, trocar mais figurinha.
- Formalismo para garantir que tarefas multidisciplinares envolvam todos os atores, nem que seja uma apresentação. Idealmente, durante o projeto para dar transparência e trocar conhecimento entre participantes
- **Not considering about the other discipline is a source of problems**
 - Tá relacionado a especificação sistêmica / alto nível. A parte da simulação AMS ... Faltou essa parte de interagir mais, trocar mais figurinha. Isso deve refletir sendo uma preocupação desde o início no projeto, não podendo ser deixado pro final pois sempre tem causado problemas e não fica tão bom quanto poderia.
- **Missing top-level view**
 - falta uma visão (alguem usar o chapéu) no nível sistêmico sem diferenciar disciplinas, focando na especificação sem se preocupar com implementação
 - visão do topo de forma mais abstrata sem se preocupar com a implementação e alinhar melhor as coisas concorrentes para refletir a expectativa do cliente e da empresa.

03 - Digital 02 ([Web view](#))

- **Missing top-level view**
 - Acaba atrapalhando muito essa falta de conversa inicial, de planejamento, de cronograma e digital não estar a par da conversa.

Formalization

segunda-feira, 6 de setembro de 2021 14:56

(Merged "No standard")

[Memo 02 \(Web view\)](#) :

- **Interactions not formalized**
 - "No standard is available for mixed-signals"

[01 - Manager 01 \(Web view\)](#):

- **No alignment of expectations on what, how and why of activities**
 - floorplan (abstract view em formato LEF). Nessa parte, eu acho que é preciso saber o que vai ser feito do outro lado e formalizar as definições.)
 - "Eu sei que tem que fazer isso", "Eu também sei", "Eu também sei", mas eu acho que falta formalizar isso.
 - ... tem falta de clareza do que significa o que foi pedido, de como tem de ser gerado, não sabia pra que ia usar e mais para esse lado... acho que essas formalizações ajudam.
 - Padrão de informação me parece mais importante que templates ou essas coisas, ... saber como tem de ser feito e quais pontos tem de ser cobertos... o que tem que ter lá dentro ou qual a precisão.
- **Missing guide for deliveries to other disciplines ("hand-off points")**
 - Padronização carece: ticket, onde se trocar as coisas (SVN, SOS e Redmine já melhorou bastante a localização de arquivos), mas pode vir de qualquer jeito: Google Docs, algo sem template, etc... Falta padronizar esses pontos de hand-off, que são os pontos chave onde se passa a bola.

[02 - Digital 01 \(Web view\)](#)

- **No alignment of expectations on what, how and why of activities**
 - Os times deveriam fazer uma conversa para ficar bem claro esse processo e conseguir ver o que tá sendo usado, em que documento isso tá registrado e assim vamos convergir.
 - Comunicação entre times geralmente é tranquila na parte informal, porém não tem um mecanismo formal para garantir que todos entendam o processo. Essa informalidade acaba focando em detalhes sem se atentar ao todo
 - As vezes gera uma demanda, um esforço do outro lado enquanto só era necessario saber se o valor estava abaixo de um determinado nível, por exemplo.
 - Sempre é importante entender o contexto dos dois lados, a informalidade sempre faz isso ficar de lado e sempre causa perda de informação.
 - A falta de formalismo reflete uma falta de contexto.
 - Talvez ter reuniões pré-definidas, checkpoints/checklists de informações a serem passadas... esse formalismo que vai ajudar nessas questões. Noto que falta trocar informações entre o time, geralmente caindo em uma conversa informal quando necessário.
 - Formalismo para garantir que tarefas multidisciplinares envolvam todos os atores, nem que seja uma apresentação. Idealmente, durante o projeto para dar transparência e trocar conhecimento entre participantes
- **Register information**
 - "O melhor não é perguntar para o projetista, é ter um acordo de onde fica registrado".

[03 - Digital 02 \(Web view\)](#)

- **Different standards accross teams**
 - O padrão utilizado pelo analógico era totalmente diferente do usado pelo digital
- **Informal communication**
 - Todo pedido de mudança era boca a boca.

Late action

sexta-feira, 27 de agosto de 2021

10:28

(Merged "No Planning")

[Memo 02 \(Web view\)](#)

- no planning for mitigation

[01 - Manager 01 \(Web view\)](#)

- **Late integration**
 - blocos de outra disciplina geralmente são considerados como caixas pretas durante muito tempo, por exemplo um bloco digital geralmente fica como black-box, ou seja, ignorado por muito tempo em um bloco superior analógico.
 - ... não existe um modelo, acaba esperando um RTL ou a netlist postlayout para ser enviado e considerado pela disciplina analog. ... temos deixado para se preocupar com ... funcionalidade... bem lá pro final do projeto.
- **Late planning, lots of rework**
 - "OK, o digital vai ser esse quadrado", mas são definições muito incipientes: quais as posições dos pinos? Quais os nomes dos pinos?
 - Para refinar o que tem de ser feito, geralmente tem muitas interações. Como geralmente ocorrem muito tarde, acaba tendo que sacrificar alguma coisa (qualidade), como checagens ou não cobrir algo.
 - ... refazer por falta de planejamento: "Como vai ser a alimentação desse bloco?" não foi conversado antes e vai ter que refazer, mas não porque alguém não fez direito. Nem tudo da para planejar inicialmente, mas é importante planejar iterativamente para limpar mais essas coisas de refazer.
 - Se o projeto puder sacrificar precisão para ser mais rápido ou precisa investir mais por ser uma questão delicado, deveríamos decidir isso de ser uma forma mais predeterminada (projetista fica sem saber, fazem o que entenderam e geralmente tem de refazer algo).
- **Late verification**
 - Features simples que você tá cuidando desde o início (na integração, na verificação) é mais difícil de deixar um buraco.
 - Bug eu não sei se já foi para silêncio, corremos para resolver e fica sem requisitos essenciais não atendido. ... pego apenas com verificações de topo muito tardias mesmo sendo questões simples.

[02 - Digital 01 \(Web view\)](#)

- **Late interaction**
 - Tá relacionado a especificação sistêmica / alto nível. A parte da simulação AMS ... Faltou essa parte de interagir mais, trocar mais figurinha. Isso deve refletir sendo uma preocupação desde o início no projeto, não podendo ser deixado pro final pois sempre tem causado problemas e não fica tão bom quanto poderia.

[03 - Digital 02 \(Web view\)](#)

- **Late verification**
 - O padrão utilizado pelo analógico era totalmente diferente do usado pelo digital, e quando notamos isso já era em cima da hora então só encapa isso aí e segue o jogo.
 - A verificação começou bem tarde, só deu tempo de fazer testes diretos. Em outro, eu já entrei no incêndio, fora do tempo... Verificação nem foi finalizada... Não deu nem tempo de fazer verificação postlayout, apenas RTL. Em outro começamos bem adiantado, mas quando voltei pra verificar a outra parte já tava na correria e tinha que aumentar o ambiente com não só novas funcionalidades, era um mundo completamente diferente com modos de operação.
- **Late integration and interaction**
 - Apesar de ter uma especificação inicial bem redondinho, com planejamento bem

arrumadinho, mas o digital entrou só nos finalmente. Sempre na cabeça do analógico (e até tem um pouco de verdade nisso) eles começam o trabalho bem antes porque a parte analógica é mais complicada, então deixam o digital pra depois que é mais simples. Digital entrou bem depois, começou já com a corda no pescoço com uma spec bem fechada.

Process

sexta-feira, 27 de agosto de 2021 10:55

[Memo 01 \(Web view\)](#) :

- low priority on top verif

[01 - Manager 01 \(Web view\)](#)

- **Specific project flow understanding helps**
 - o Acho que ter uma **visão do fluxo específica do projeto** corrente bem cristalizado ajudaria na clareza, pois acho que não fica claro a **utilidade de cada arquivo** para todos.
 - o Que recebemos do analog, geralmente é **Liberty e LEF (abstract)**, talvez alguma coisa para simulação digital como um **modelo Verilog** do analogico, mas geralmente nós geramos ou temos algum envolvimento (nunca geramos um modelo sem auxílio nosso).
 - o Ainda no LIB, se o projetista analog **não tem essa clareza ou a ferramenta** (scripts) para isso, o projetista digital **precisa gerar (ou pedir) o que ele precisa usar** ao invés de apenas receber.
 - o ... ECO novamente por causa de **problema com pinos (pinout errado** em arquivos)... **não é uma preocupação** pro projetista analog ("**botei qualquer direção**")... quando traçamos o problema, **era no LEF recebido** pelo time digital que um pino estava com uma **propriedade errada**.
 - o Se o projeto puder sacrificar precisão para ser mais rápido ou precisa investir mais por ser uma questão delicada, deveríamos **decidir isso de ser uma forma mais predeterminada** (projetista fica sem saber, fazem o que entenderam e geralmente tem de refazer algo).
- **Missing or incomplete information sharing**
 - o ... pois não basta saber **o que o sistema faz**, também é importante saber **até onde vamos modelar, ...** É importante fazer esse **julgamento dessas questões**. Acho que **falta (nós, os gerentes)** conversarmos mais para os projetistas entenderem o que precisa nesse modelo. As vezes **não demos subsídios suficientes** para o projetistas saber o que tem que modelar.
- **Inspection instead of prevention approach**
 - o **"Eu to te passando, se der erro me avisa"**, mas acabamos nos complicando nisso.
 - o **Para refinar** o que tem de ser feito, geralmente tem **muitas interações**.
- **Not a "people problem"**
 - o Em termos de disponibilidade (o outro time) está ok (não mostra desinteresse ou ineficiência, algo assim)
 - o Quando faço um pedido para o analog, eu acho que as coisas se resolvem num tempo bom e de boa forma.
 - o Acho que é mais a questão de **refazer por falta de planejamento**
 - o ... como garantimos que todos tem **acesso a informação** de como tem que fazer, como tem que entregar?

Essa fala ressoa muito com Total Quality...
É mais focado na auditoria de validade do que na entrega planejada para estar correta "de primeira"

[02 - Digital 01 \(Web view\)](#)

- **Versioning requires effort to implement, but brings transparency and clarity**
 - o Uma mudança de **ferramenta de versionamento** foi transparente pro digital, mas apresentou **dificuldades pro analog**
 - o **ferramenta de versionamento, tendo resistencia na adoção desta e de uma política de uso social**.
 - o Essa **convergência não é tão fácil para o time analógico**, apesar de já ter avançado bastante com as pessoas acostumando. É um trabalho gradual.
 - o Também entra as **ferramentas de versionamento para deixar mais transparente**,

[03 - Digital 02 \(Web view\)](#)

- **Outdated documentation**
 - o A entrega foi feita por SVN e indica qual revision era, **não teve documentação ou requisição referente a essa mudança**.
 - o Apesar de ter uma spec inicial, vem **várias mudanças e a documentação fica desatualizada**.
 - o Acaba ficando muito bagunçado, aí quando vai retomar um projeto a documentação **tá toda desatualizada**. Ai fica tentando lembrar de memória porque algumas decisões foram tomadas aí no meio de uma reunião você vai lembrando de repente as coisas. Esse replay para concertar bugs da versão anterior com documentação desatualizada tem o problema de pessoal que saiu da empresa e aí nem essa conversa dá para ser feita direito e fica **informação perdida**.
 - o **"Tem uma black box no meio do projeto", "Mas o que é isso aí? Vamos tirar", "Não tira não que tá funcionando"**.
 - o "Não é raro ter uma rerodagem", então tem que considerar que o projeto vai voltar então tem que deixar um tempo depois do projeto só para documentar ao invés de já jogar em outro projeto

Aqui conversa com aquele documento de Architecture Decision Record (ADR) que poderia trazer benefícios nessa questão

Acaba faltando um pós-projeto para deixar a "casa arrumada".

Interface

sexta-feira, 27 de agosto de 2021

10:26

Bundle of ports, usual source of integration issues
(merging "Integration issues" code)

[Memo 01 \(Web view\)](#) :

- **constant problematic integration**

[01 - Manager 01 \(Web view\)](#)

- **Not aligned early in the project**
 - OK, o digital vai ser esse quadrado", mas são **definições muito incipientes**: quais as posições dos pinos? Quais os nomes dos pinos?
- **Integration problems**
 - Nomes diferentes são usados de cada lado (**mismatch de nome de pino**) e em alguns projetos foram necessários wrappers para **adaptar os pinos** mesmo estando tudo **dentro da mesma empresa**. Também acontece pinos com **direção diferentes**, pino de **power sem propriedade de power**, entre outros.
 - ... ECO novamente por causa de **problema com pinos** (**pinout errado** em arquivos). Nele, um dos pinos digitais veio como power/ground. Como era um **bloco com muito pino**, gerou um **warning** (que passou despercebido) e **suprimiu uma lógica**. Esse erro num arquivo simples é um **tipico ponto** que **não é uma preocupação pro projetista analog** ("**botei qualquer direção**"), mas resultava num curto quando a lógica estivesse em alto. ... traçamos o problema, **era no LEF recebido** pelo time digital que um pino estava com uma **propriedade errada**.
 - Interface é a parte onde chega o que você não sabe ("**eu conheço até aqui, tô trabalhando nessa sala**"), e a mesma coisa do outro lado ("**na outra sala**"). Esse problemas mostram que "**logo ali tá ficando um ponto cego**". "**Os dois pararam de enxergar um pouco antes e ficou um buraco**".

[03 - Digital 02 \(Web view\)](#)

- **Integration problems**
 - Porém já tive **muito problema com interface**. Um problema muito grande em um projeto foi os **pinos estarem totalmente diferentes no digital e no analog**. Foi necessario fazer um wrapper de topo só pra renomear as portas e rerotiar alguns indices de fio (digital usava bit 3 e analogico usava bit 20, algo assim, principalmente bits de controle).
 - O **padrão utilizado pelo analógico era totalmente diferente do usado pelo digital**
- **Constant updates**
 - Teve projeto que **pinos da interface mudavam da noite pro dia, funcionamento mudando em cima da hora**.

Requirement

sexta-feira, 27 de agosto de 2021

11:42

01 - Manager 01 ([Web view](#))

- **Unclear requirements**

- "OK, o digital vai ser esse quadrado", mas são definições muito incipientes: quais as posições dos pinos? Quais os nomes dos pinos?
- Eu tenho noção no que as entregas são usadas pela outra disciplina, mas não acho que isso está claro para todos os envolvidos.
- Modelos nem sempre estão bem definidos, pois não basta saber o que o sistema faz, também é importante saber até onde vamos modelar, o que nem sempre está claro pros projetistas.
- O problema de comunicação geralmente mostra que não deixou claro o que tu quer receber ou o que tem de estar verificado (não tá escrito, não tem ticket, ticket muito sumariado, definição não clara).
- de vocabulário não parece ter muito problemas, mas tem falta de clareza do que significa o que foi pedido, de como tem de ser gerado, não sabia pra que ia usar e mais para esse lado... acho que essas formalizações ajudam. Não sei se é "Tu não me entendeu bem" ou "Eu não expliquei direito"

- **Low priority on requirement definition**

- Um exemplo foi um projeto onde tivemos uns problemas que ficaram meio que na zona cinza, como uma feature que é controlada pelo digital, mas o entendimento da feature é mixed pois estava relacionado com medição de corrente. Por isso, na hora de especificar e validar as features, tem que ter uma visão mixed no topo e as duas disciplinas tem que se entender (as vezes até considerar extra-chip: dependendo dos instrumentos de medição que serão usados ou testes em silício). Essa feature foi pra spec pouco detalhada e, apesar de muito simples, tinha de ser visto na perspectiva do sistema.
- Acho que isso também é o ponto de deixar muito claro no topo o porquê das coisas

- **Some requirements left out without necessity**

- Bug eu não sei se já foi para silício, corremos para resolver e fica sem requisitos essenciais não atendido. ... pego apenas com verificações de topo muito tardias mesmo sendo questões simples. Acho que isso também é o ponto de deixar muito claro no topo porque o reset precisa ser síncrono ou etc e não definir requer adaptações de lógica, reposição e reroteamento de pino.
- As vezes são features não-essenciais ou problemas que são contornáveis, então não chegou a inviabilizar o projeto.

02 - Digital 01 ([Web view](#))

- **Low priority on requirement definition**

- Por regra (99.99% dos casos), nós não temos requisitos elétricos, um handshake, os DRVs e etc... que deveríamos dar uma atenção boa.

- **Lacking requirement engineering**

- Do ponto de vista funcional, temos o problema de como é que se chega de uma spec para um circuito digital? "De onde vem a spec"?
- Acho que é mais o ponto de especificação, onde e como colocar essa spec. Tem de ser simples para ter uma clareza.
- Nessa parte de spec, é um ponto que tem de ser discutido como ser feito para se ter essa visão do topo de forma mais abstrata sem se preocupar com a implementação e alinhar melhor as coisas concorrentes para refletir a expectativa do cliente e da empresa. Specs não realistas e etc são refinadas ao longo do projeto para facilitar a interação com o cliente.

- **Some requiremnts left out without necessity**

- O que acontece é afetar a qualidade do produto final (mais no sentido de **problemas não essenciais**, como valores default ou exigir um workaround para funcionar) ou algum problema que **não conseguimos entender o motivo**.

03 - Digital 02 ([Web view](#))

- **Unclear requirements**
 - Parece indicar que, apesar de disponibilidade, não existia descrições suficientemente completas para compreensão do sistema

Negative impact

terça-feira, 7 de setembro de 2021 11:51

01 - Manager 01 ([Web view](#))

- **No show-stopper problems**

- Geralmente o impacto desses problemas é atraso, menor produtividade. Bug eu não sei se já foi para silício, corremos para resolver e fica sem requisitos essenciais não atendido.

02 - Digital 01 ([Web view](#))

- **No show-stopper problems**

- Esse problemas geralmente geram retrabalho, trabalho desnecessário ou mais complexo do que necessário, bugs tardios ou não pegos... Uma falha catastrófica não passa porque temos múltiplas etapas de verificação, nós tendemos a pegar eles apesar de poder acontecer. O que acontece é afetar a qualidade do produto final (mais no sentido de problemas não essenciais, como valores default ou exigir um workaround para funcionar) ou algum problema que não conseguimos entender o motivo.

03 - Digital 02 ([Web view](#))

- **Mental health**

- "Pelo menos a parte de verif, nunca tive uma experiência boa com a Chipus, sempre correndo pra apagar incêndio".
- Pensando do meu lado, não é desconfortável... é uma palavra mais negativa que isso... Aquele sentimento de que não tá fazendo trabalho bem feito, porque é correria, estresse... quer colocar o que tu aprendeu na prática mas não dá na vida real, vai o que der. Saúde mental do trabalhador: cara vai dormir e fica pensando naquilo, acaba sendo muito desgostoso.
-

[[OBSERVATIONS]]

segunda-feira, 6 de setembro de 2021 15:49

The development is divided in 3 macro **phases** according to the abstraction level of the circuit: **functional** (behaviour), **electrical** (transition times and loads) and **geometry** (backend and layout). Those phases are perceived with different **discipline visions** according to the actors: digital vision, analog vision, mixed-signal / top-level / systemic vision and extra-chip (like workbench or in-field operation). The most mature interactions are related to geometry artifacts, which have a more structured procedure and, consequently, less unnecessary overhead even though there's still a constant back and forth.

Even though all essential **requirements** are ok at the end, few non-essential ones usually are left out or workarounds are required to operate the IC as intended. The main problem seems to be related with the **interface** between digital and analog, where a shared (systemic) vision is not considered, constant changes occur and teams have divergent nomenclature standards.

During most of the project, the other team block is considered a black box with very low specification or consideration for the other teams flow and context. It seems that both teams "stop looking" a little before the interface, creating a grey-area gap. More transparency and integration of the designers would mitigate and share knowledge about the project, improving the overall quality.

It's suggested that the lack of **formalization** is the main reason for this incompatibility between the technical flows. The usual informal exchange of information focus on specific details without considering the systemic vision, missing important context for optimal procedures. **Process** of development would have less unnecessary work and more complete deliverables considering context, but must also include time to register design/architecture decisions and update documentation for future project versions.

This formalization would bring practices to guarantee the transparency needed for a better contextualization of the process and increase the focus for a more complete specification earlier in the project. A better understanding of steps and tools of the other team will make inter-discipline deliveries more consistency and reduce the necessity to correct them. The standard process documents of Chipus should be improved to reflect established geometry procedures and create missing formalizations to guarantee those key points (verification, planning, specification, integration, project flow) early in the project, focusing on prevention of problems instead of inspection.

The **negative impacts** in the project are usually schedule and non-essential requirements. Show-stoppers that make the product invalid don't occur, but quality-of-life features often are missing or workarounds are required for the correct usage of the chip. Designer also suffer negative impacts regarding mental health, not only feeling frustrated for not being able to use learned skills or having to trade off quality because of time, but also constant stress caused by the tight schedules and instability of what must be done.



