

INSTITUTO FEDERAL

Santa Catarina

Câmpus Tubarão

Introdução a Sistemas Embarcados

Curso Superior de Tecnologia em Sistemas Embarcados

Professor: Fernando Silvano Gonçalves

fernando.goncalves@ifsc.edu.br

Fevereiro de 2023

Cronograma

Encontro	Data	Nº Aulas	Conteúdo
1	7-fev.	04	Recepção e Apresentação do Unidade / Apresentação do Plano de Ensino / Avaliação Diagnóstica / Introdução a sistemas embarcados / Conceitos, Características e Aplicações
2	14-fev.	04	Visita Técnica Evolumenta Sistemas
3	28-fev.	04	Histórico de Sistemas Embarcados / Conceitos de Projeto de Sistemas Embarcados
4	7-mar.	04	Conceitos de Projeto de Sistemas Embarcados
5	14-mar.	04	Conceitos de Projeto de Sistemas Embarcados
6	21-mar.	04	Introdução à Linguagens de Programação
7	28-mar.	04	Variáveis e Operadores
8	4-abr.	04	Estruturas Condicionais
9	11-abr.	04	Estruturas de Repetição
10	18-abr.	04	Avaliação 01



Cronograma

Encontro	Data	Nº Aulas	Conteúdo
11	25-abr.	04	Microcontroladores
12	2-mai.	04	Entradas e Saídas Digitais
13	9-mai.	04	Conversor Analógico-Digital
14	16-mai.	04	Sensores
15	23-mai.	04	Comunicação Serial
16	30-mai.	04	PWM
17	6-jun.	04	Temporizadores
18	13-jun.	04	Interrupções
19	20-jun.	04	Avaliação 02
20	27-jun.	04	Conselho de Classe / Atividades de Encerramento da UC
		80	



Resumo

- Introdução a Sistemas Embarcados;
- Conceitos e Características;
- Aplicações;

Pauta

- Evolução dos Sistemas Embarcados;
- Complexidade de Sistemas Embarcados;
- Processo de Desenvolvimento de Sistemas Embarcados;

Evolução dos Sistemas Embarcados

Histórico dos Sistemas Embarcados

1960: Primeiro Sistema Embarcado:

- Apollo Guidance Computer.



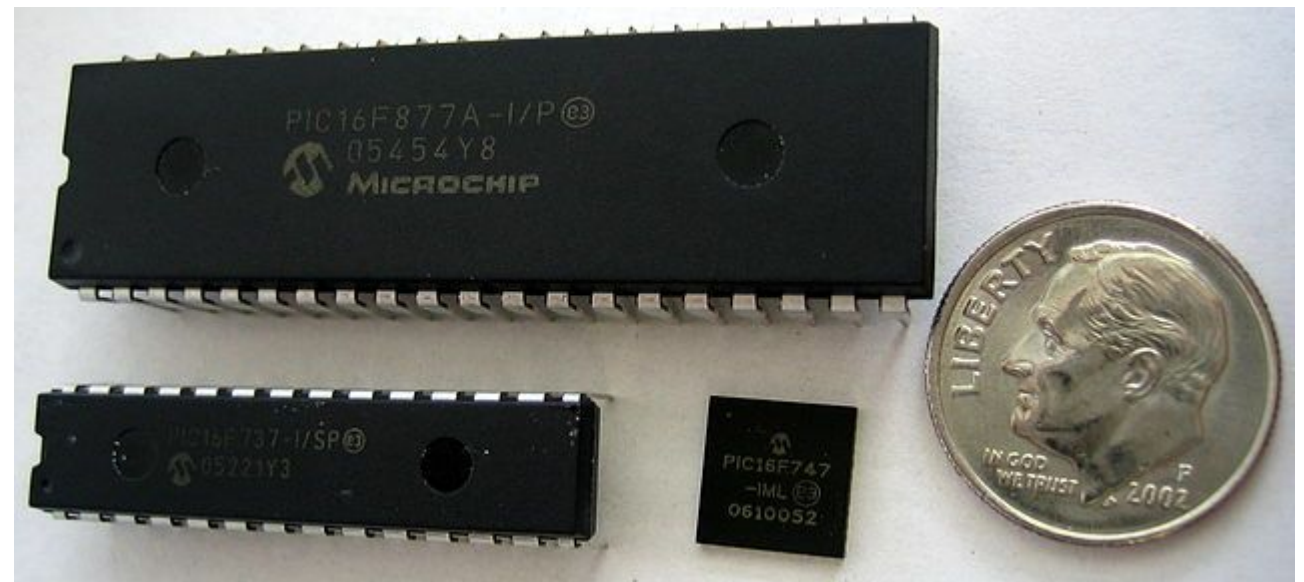
Histórico dos Sistemas Embarcados

- ❑ **Até 1978:** Utilizado basicamente em aplicações militares;
 - Controle de radares e mísseis.



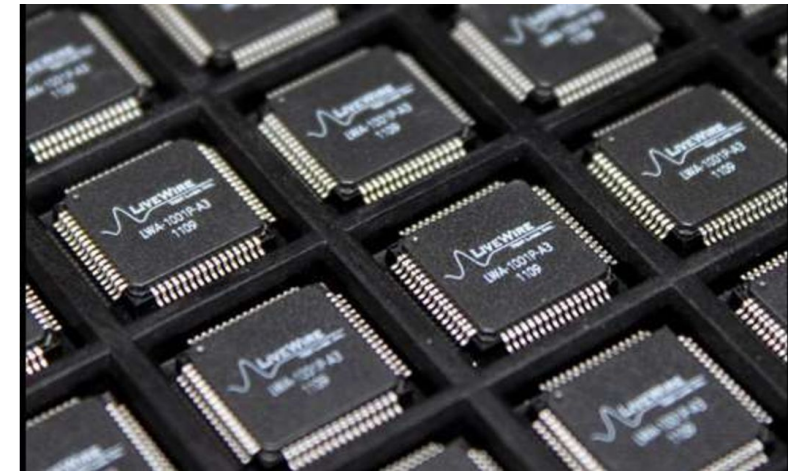
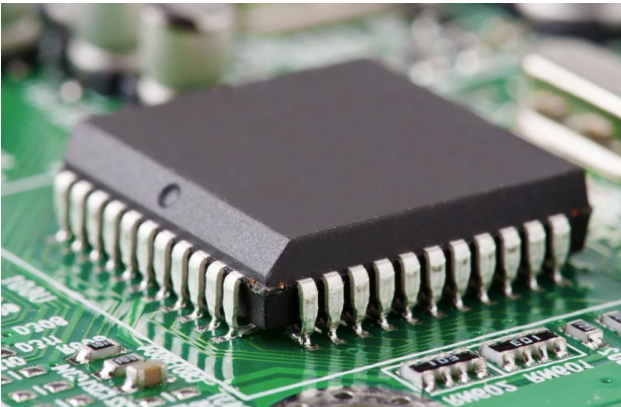
Histórico dos Sistemas Embarcados

- ❑ **1984:** Explosão dos microcontroladores:
 - Microprocessadores otimizados para controlar dispositivos eletrônicos;
 - Memória e interfaces de E/S integrados enfatizando a autossuficiência.



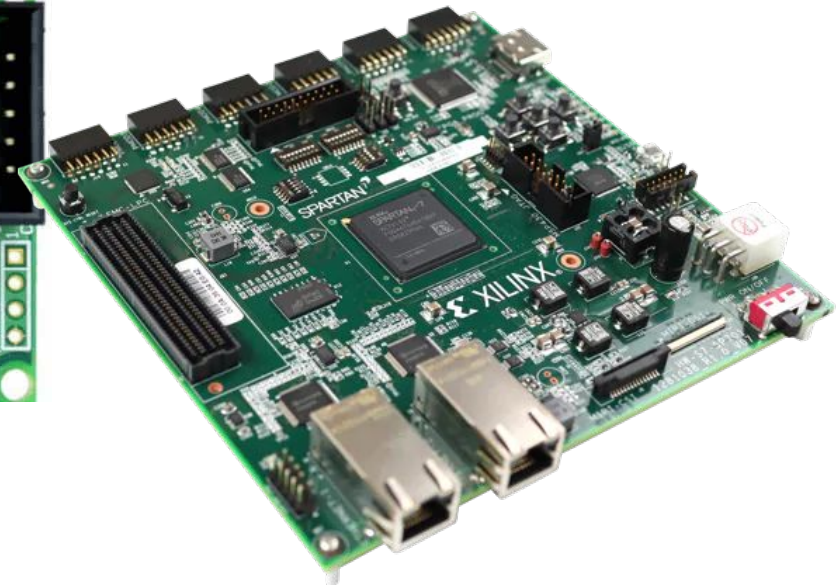
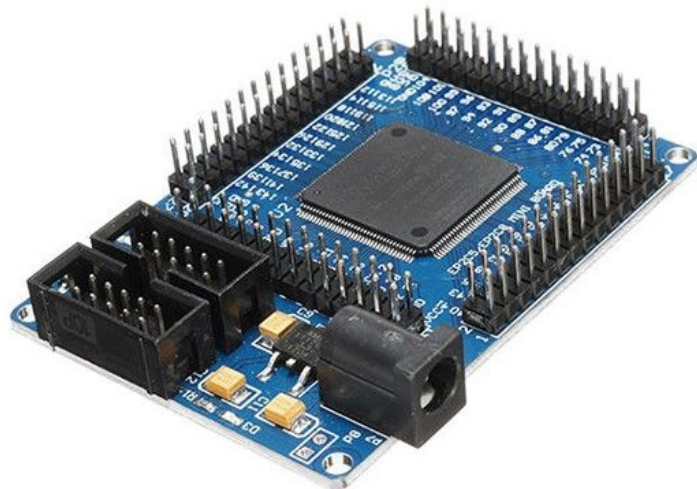
Histórico dos Sistemas Embarcados

- ❑ **1989:** Popularização dos Application-Specific Integrated Circuit (ASICs);
 - Descritos pelas Hardware Description Language (HDL), como Verilog ou VHDL;
 - Ferramentas de Síntese;



Histórico dos Sistemas Embarcados

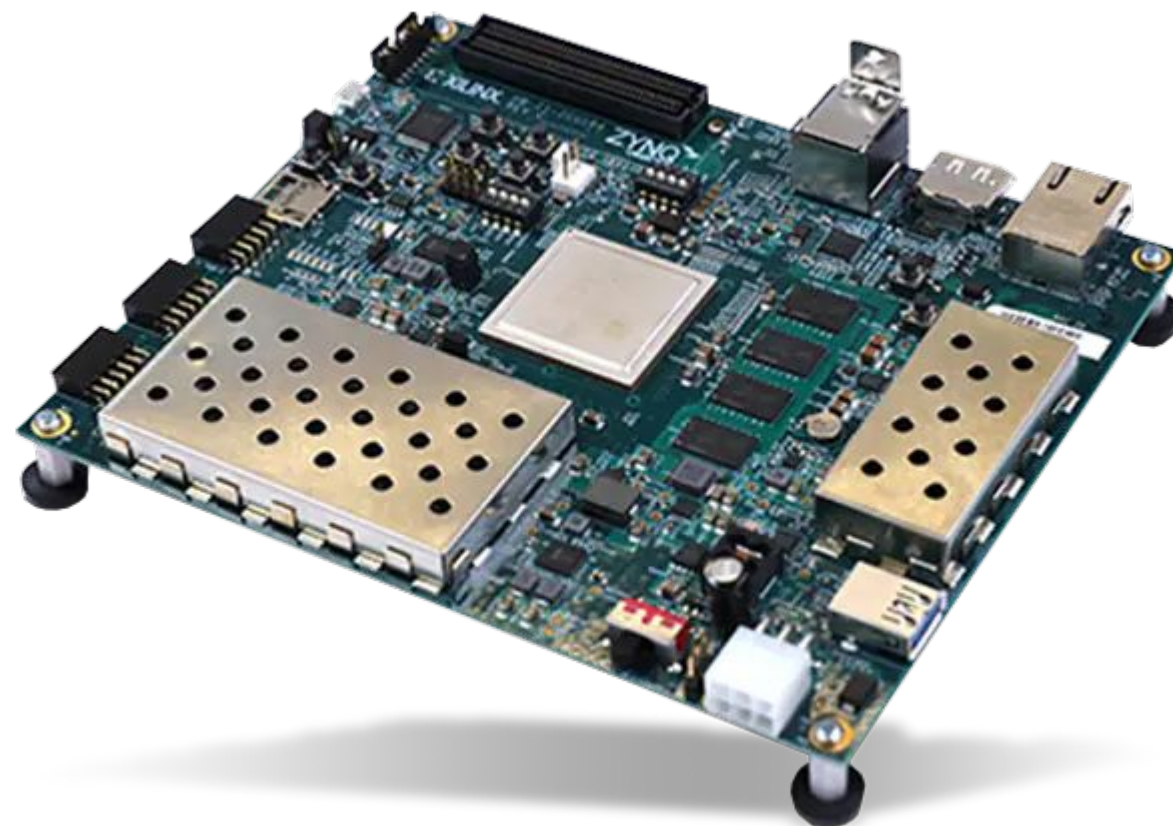
- ❑ **1995:** Criação das Field-Programmable Gate Arrays (FPGA);
 - Revolução dos System-on-a-Chip (SoCs): CPU e periféricos integrados com circuitos eletrônicos em um único chip;

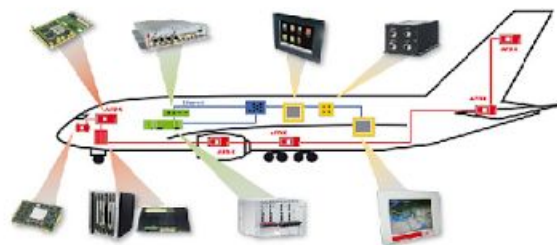
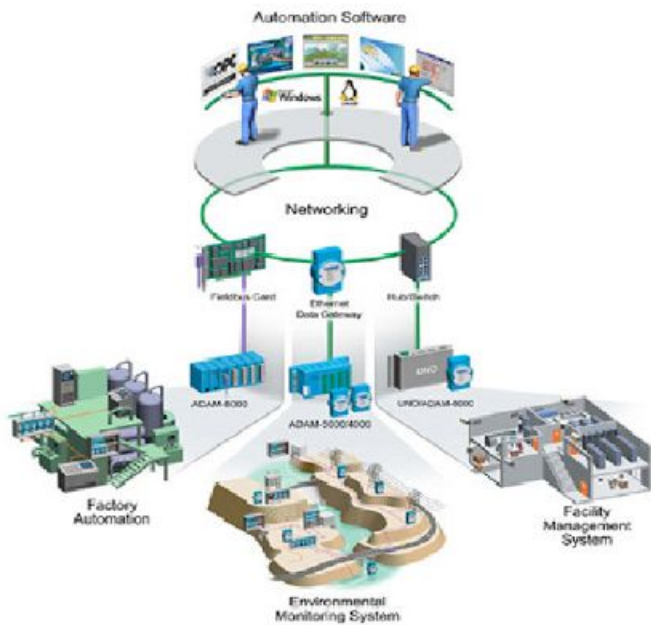


Histórico dos Sistemas Embarcados

❑ **Anos 2000:** aumento da capacidade e redução dos custos dos FPGAs;

- **MPSoC:** Multi-processors SoC;
- **NoC:** Network-on-Chip;
- **Sistemas Reconfiguráveis;**





Por que Projetar um SE é um Processo Complexo?


- ❑ Custo limitado:
 - ❑ Normalmente agregado a outro produto;
- ❑ Tempo de projeto limitado:
 - ❑ Projetos muito longos podem significar perdas de mercado;
- ❑ Consumo de energia:
 - ❑ Geralmente movidos por fontes limitadas (baterias);
- ❑ Complexidade crescente das aplicações:
 - ❑ Ambientes exigem cada vez mais aplicações seguras e confiáveis.

Dificuldades de Projeto de SE

Lei de Moore:

-  Número de transistores disponíveis nos circuitos dobra a cada 18 meses;

Custos de engenharia não recorrentes:

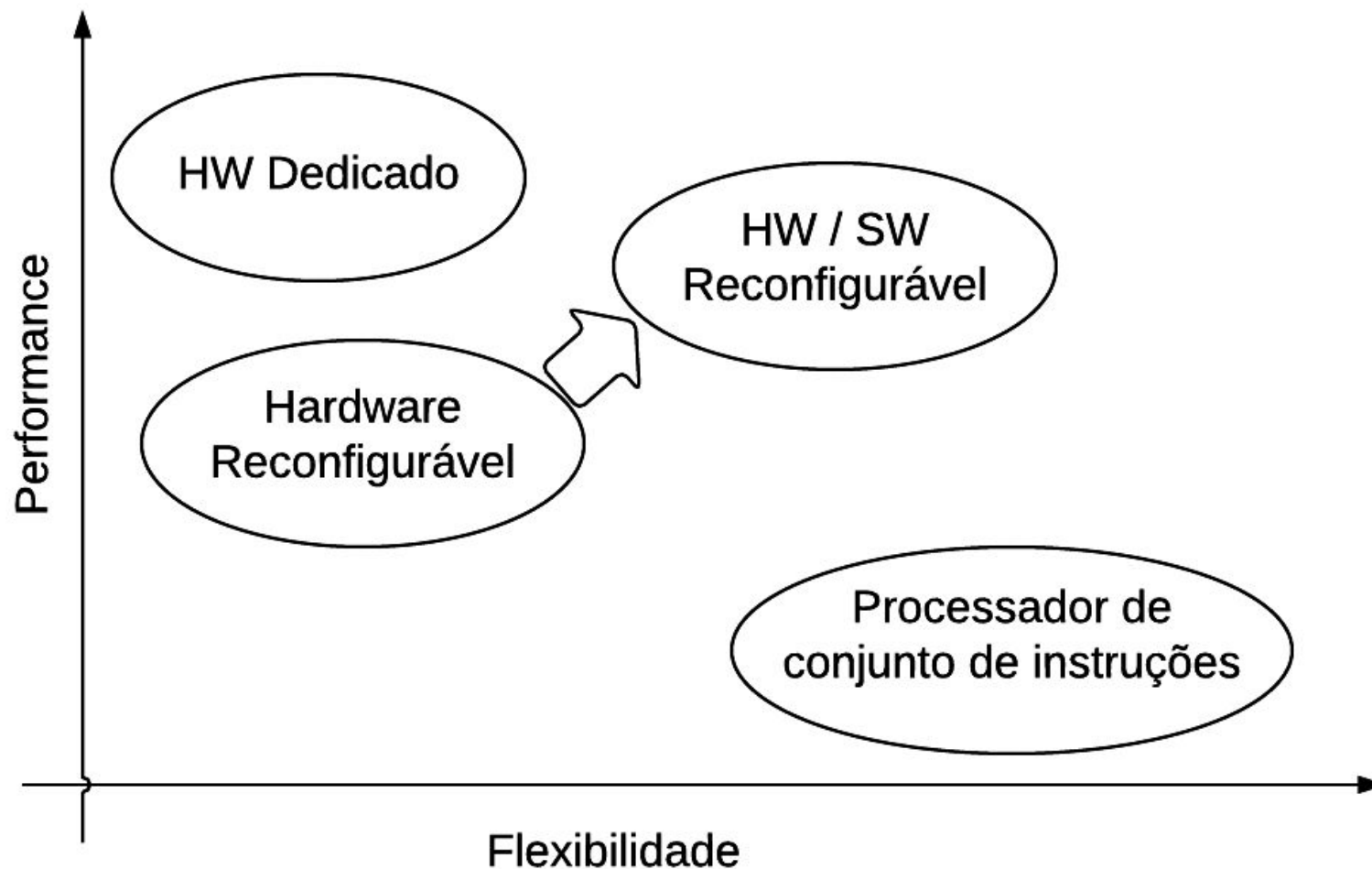
-  Custo para preparar uma fábrica para produção de um determinado chip (próximo de USD 1 milhão).

Como Projetar um SE?

☐ Análise detalhada do problema detalhando suas necessidades NFRs:

- ☐ Desempenho;
- ☐ Consumo de energia;
- ☐ Flexibilidade;
- ☐ Custo;
- ☐ Conectividade.

Performance VS Flexibilidade



Como Projetar um SE?

- ❑ Dominar tecnologias disponíveis:
 - Arquitetura de processadores (CISC, RISC, DSP);
 - Modelos de computação;
 - Flexibilidade;
 - Linguagens de programação (HW e SW);
 - Ferramentas de desenvolvimento;

Atividade MULTIDISCIPLINAR!

Como domar essa Complexidade?

Como domar essa Complexidade?

Como domar essa Complexidade?

- ❑ Modelagem;
- ❑ Metodologias de Projeto;



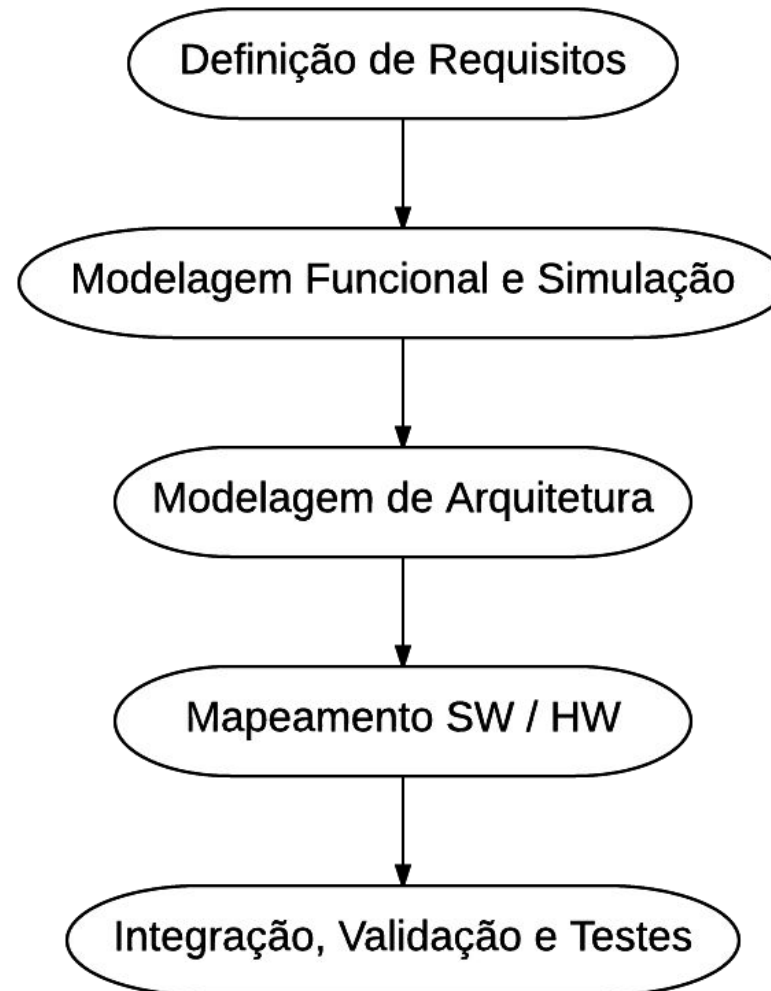
Como domar essa Complexidade?

❑ Uso de Metodologias de Projeto adequadas:

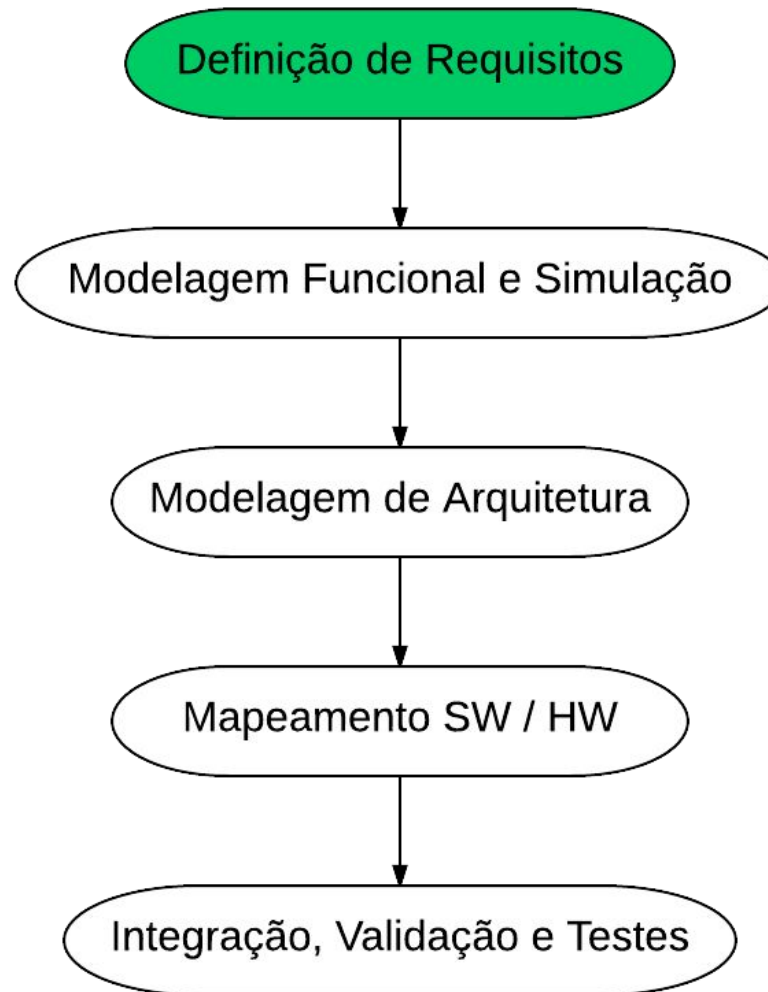
- Aumenta o nível de abstração;
- Reduz falhas de especificação;
- Matem o "conhecimento corporativo";
- Aumento nos índices de produtividade;
- Aumento do reuso de software e hardware.

Processo de Desenvolvimento de Sistemas Embarcados

Processo de Desenvolvimento de SEs



Processo de Desenvolvimento de SEs



Definição de Requisitos

☐ Requisitos funcionais:

- Descreve as funcionalidades que o sistema deve realizar;

☐ Requisitos não funcionais:

- Apresentam as restrições impostas ao sistema para realização dos requisitos funcionais.

Características dos Requisitos Não Funcionais

- ☐ Restrição temporal;
- ☐ Determinismo;
- ☐ Tolerância a falhas;
- ☐ Segurança;
- ☐ Interface;
- ☐ Custo;
- ☐ Etc.

Requisitos Funcionais e Não Funcionais

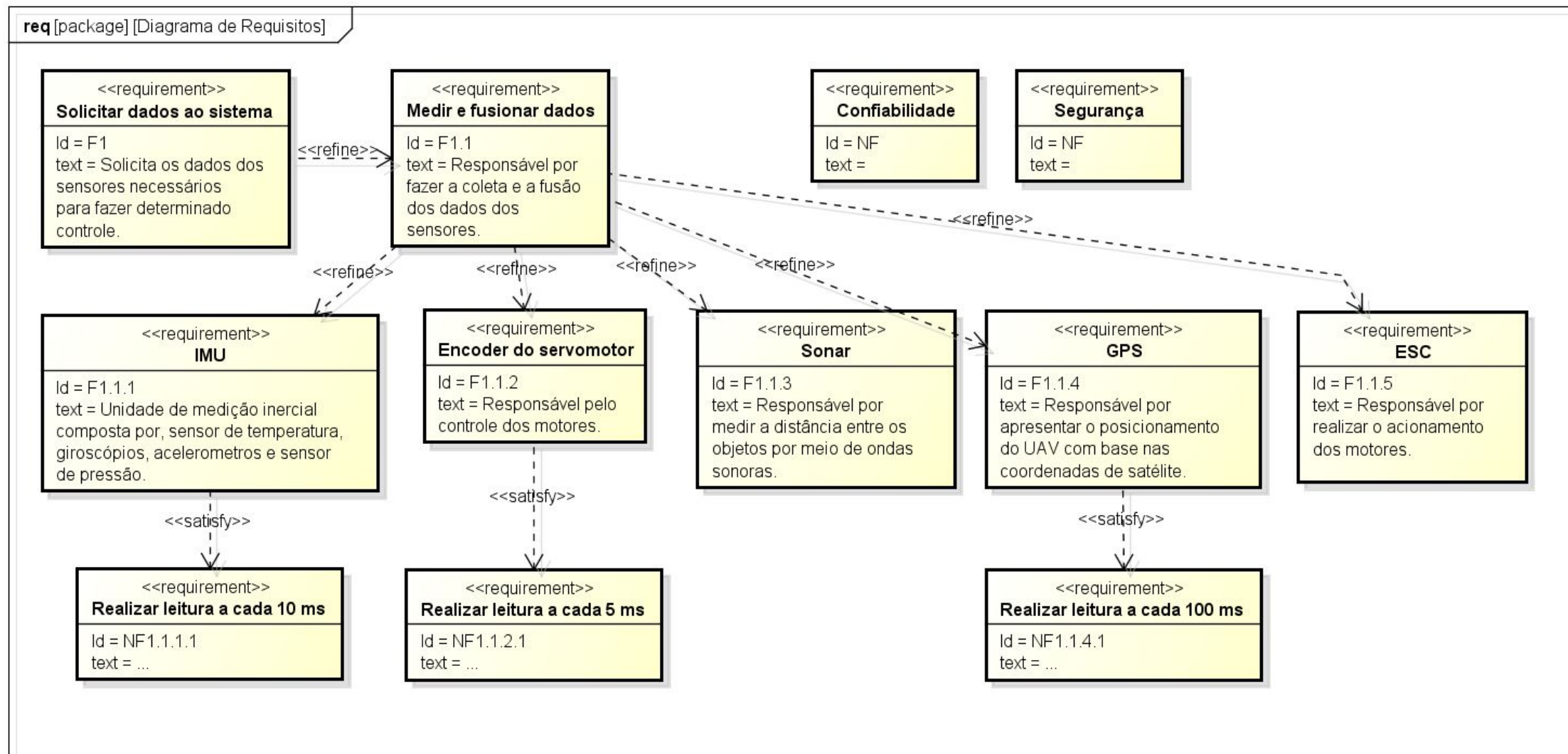


Tabela de Requisitos Funcionais

- ☐ Código do Requisito Funcional;
- ☐ Nome do Requisito;
- ☐ Descrição;
- ☐ Categoria Funcional.

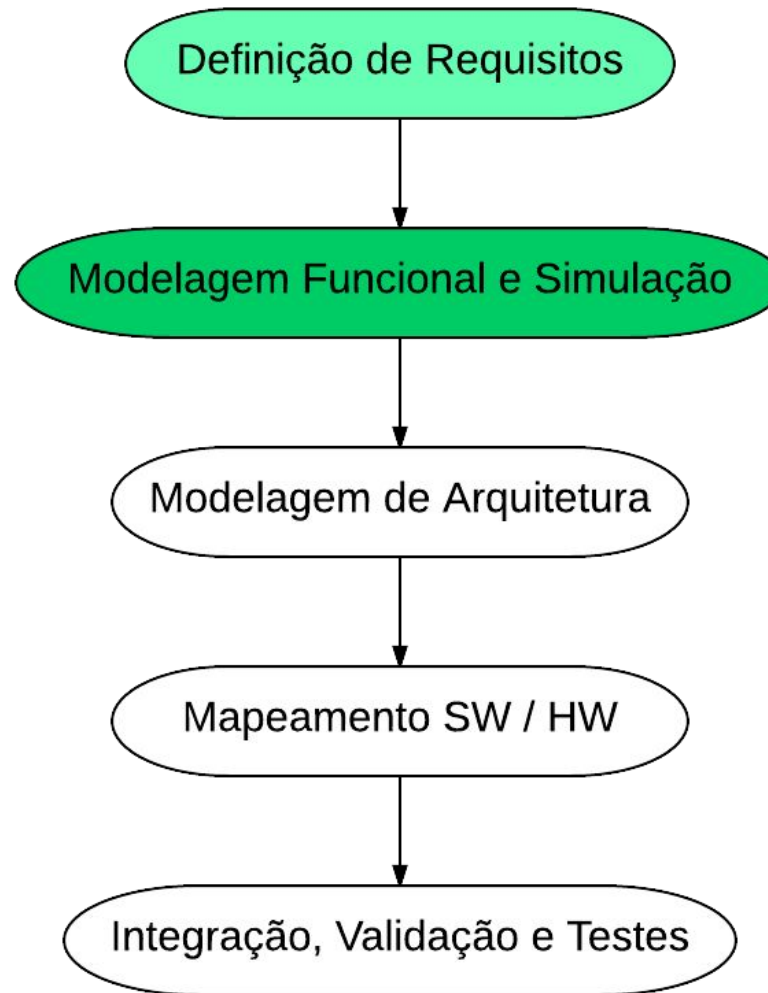
Tabela de Requisitos Não Funcionais

- ☐ Código do Requisito Não Funcional;
- ☐ Nome do Requisito;
- ☐ Restrição;
- ☐ Categoria;
- ☐ Obrigatoriedade;
- ☐ Permanência.

Tabela de Requisitos

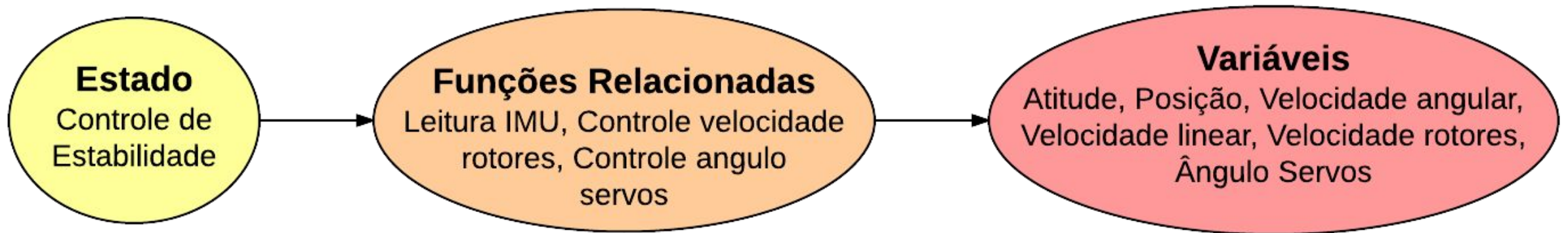
F1 - Controle de Estabilidade em Voo				Oculto ()
Descrição: O sistema deve manter a estabilidade da aeronave durante a realização dos voos, sejam estes autônomos ou rádio controlados.				
Requisitos Não Funcionais				
Nome	Restrição	Categoria	Tempo Real	Permanente
NFR1.1 - Inclinação Máxima lateral	A aeronave não pode ultrapassar uma inclinação máxima de 20° no angulo de rolagem	Segurança	Hard	(X)
NFR1.2 - Nível Mínimo de Bateria	A aeronave não pode realizar voos se o nível de tensão da bateria estiver abaixo de 14v.	Segurança	Hard	(X)
NFR1.3 - Alerta Sonoro	Se o nível de tensão da bateria estiver abaixo de 14v, a aeronave deve emitir um aviso sonoro.	Interface	Soft	(X)

Processo de Desenvolvimento de SEs



Modelagem Funcional

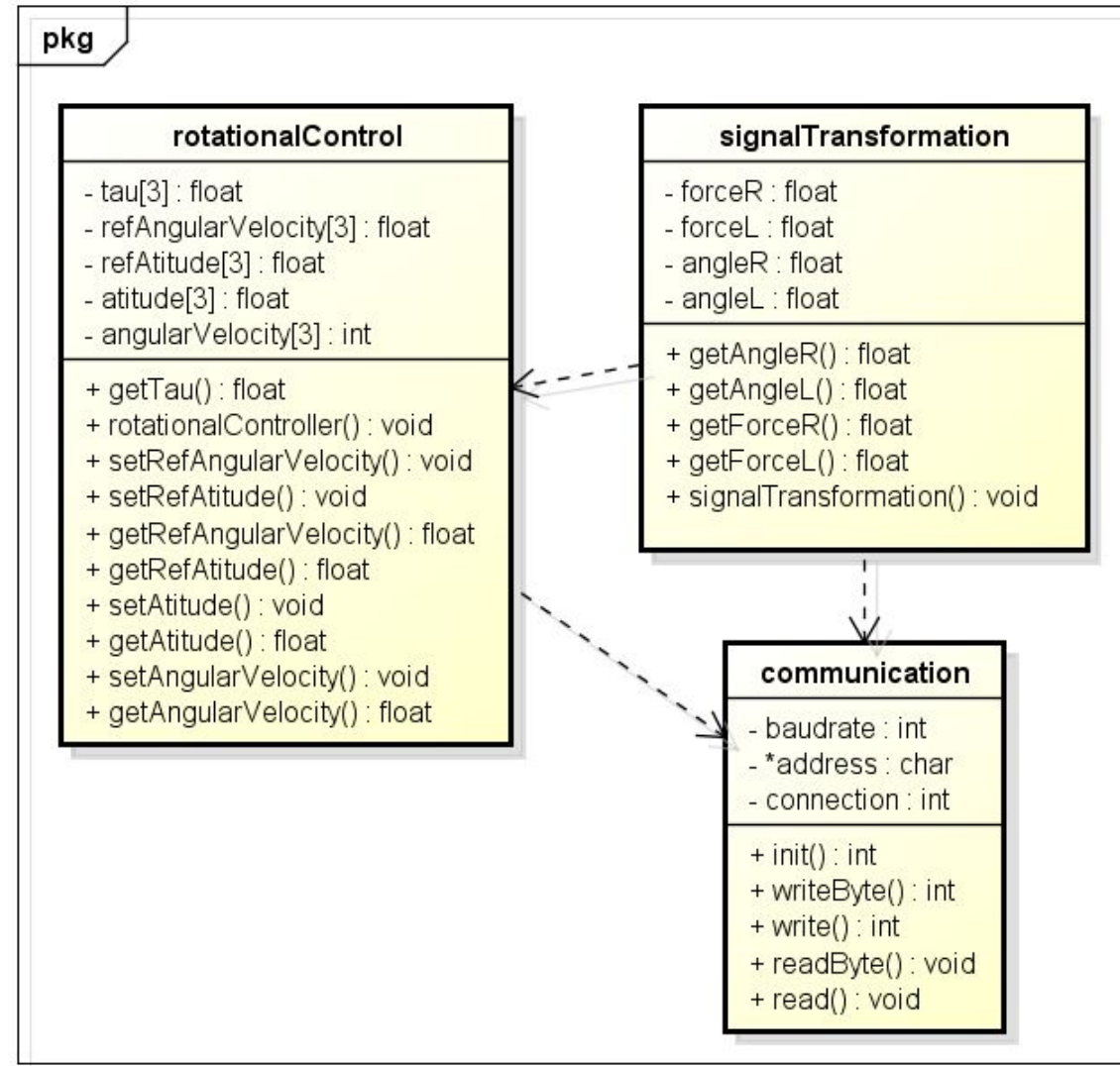
- ❑ Expressa as **funcionalidades** de maneira abstrata;
- ❑ **Independente** de plataforma;
- ❑ Identificação dos **Estados, Funções e Variáveis Relacionadas**;



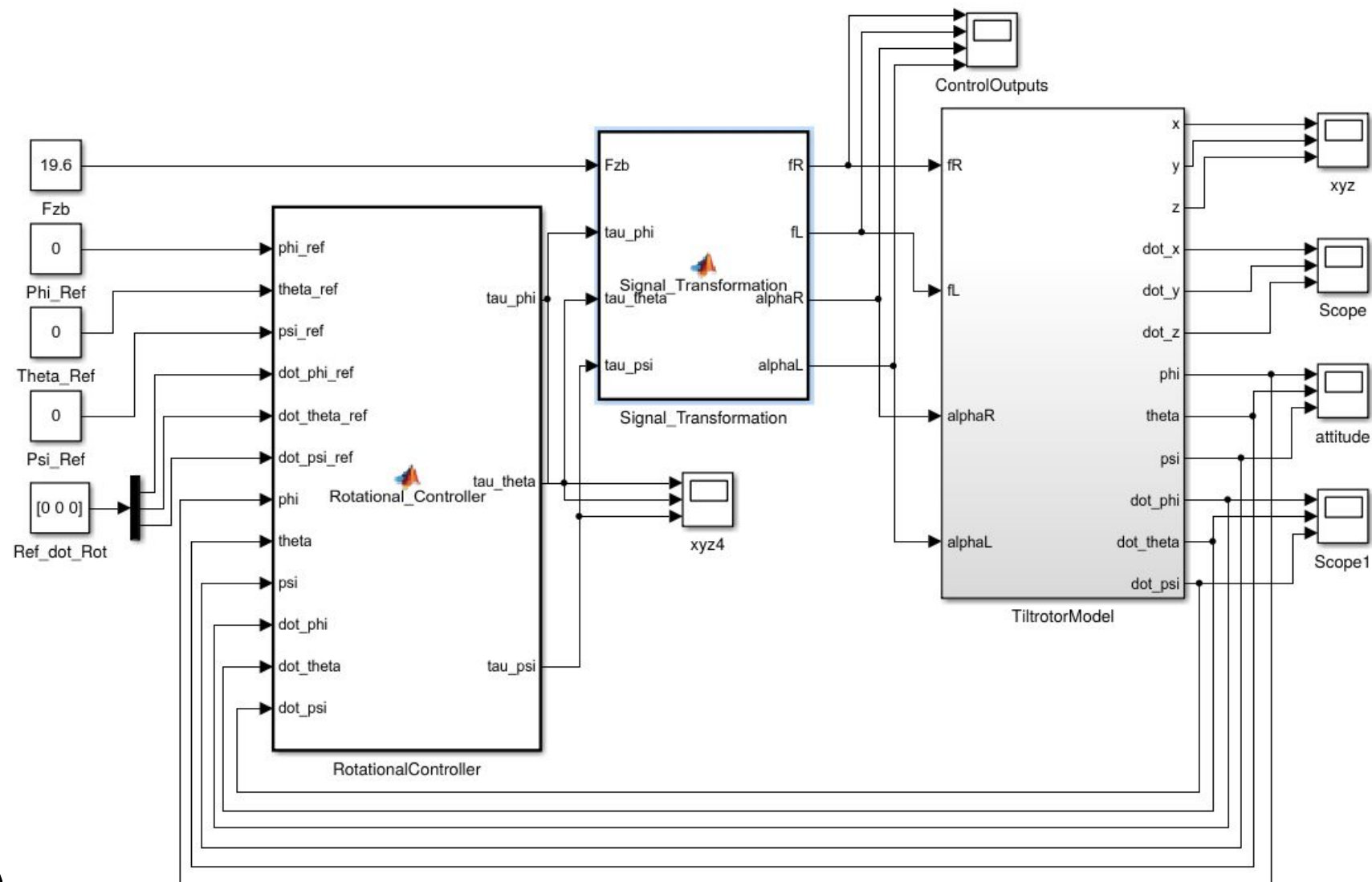
Modelagem Funcional

- ❑ Modelos refletem o **comportamento** do sistema;
 - Identificação dos modos de operação;
 - Escolha de uma linguagem e ferramenta para cada domínio;
- ❑ **Modelos Contínuos:**
 - Blocos Funcionais (Simulink, Ptolemy);
- ❑ **Modelos a Eventos Discretos:**
 - State flows, Statechart (Simulink, Rational-IBM);

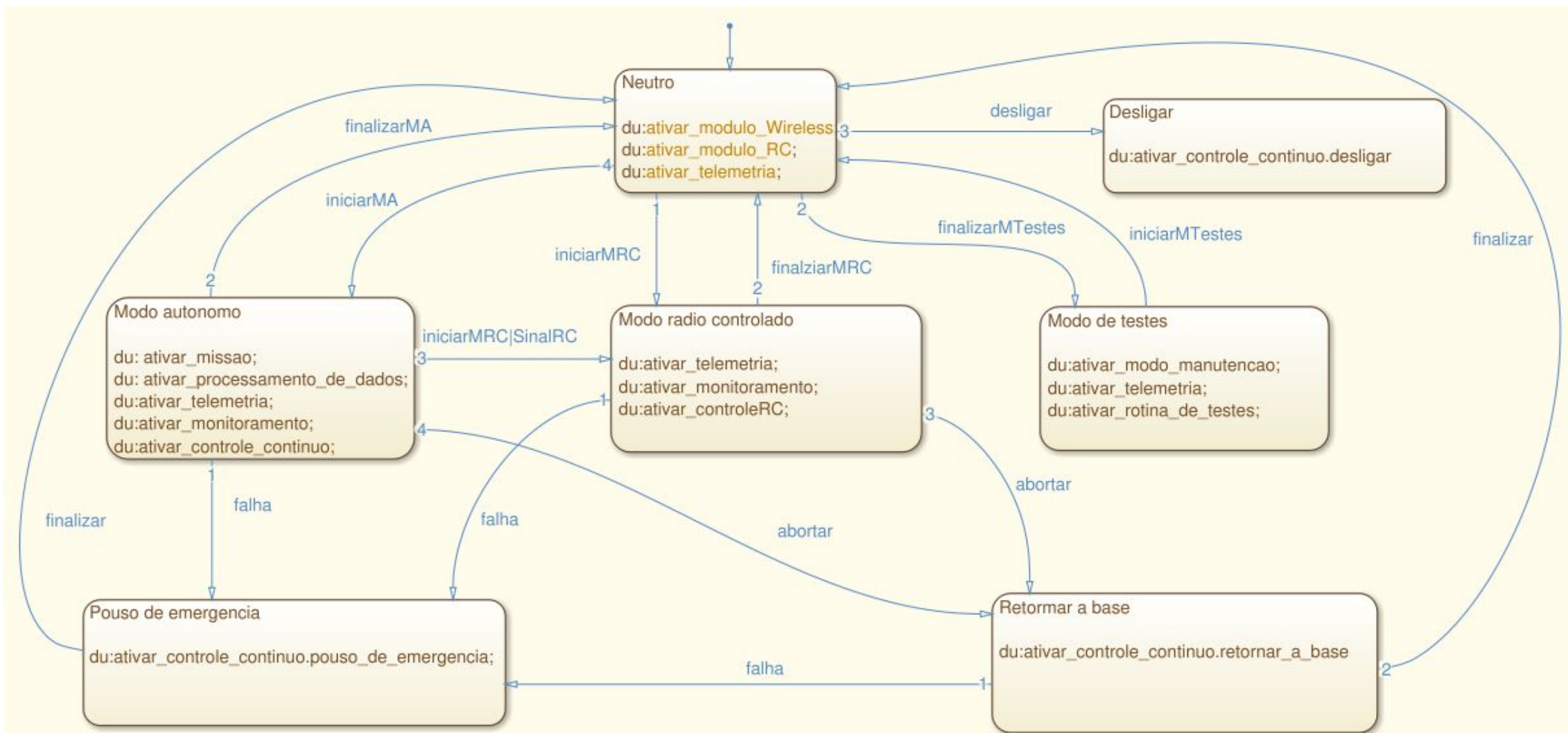
Modelagem Funcional - Diagrama de Classes UML



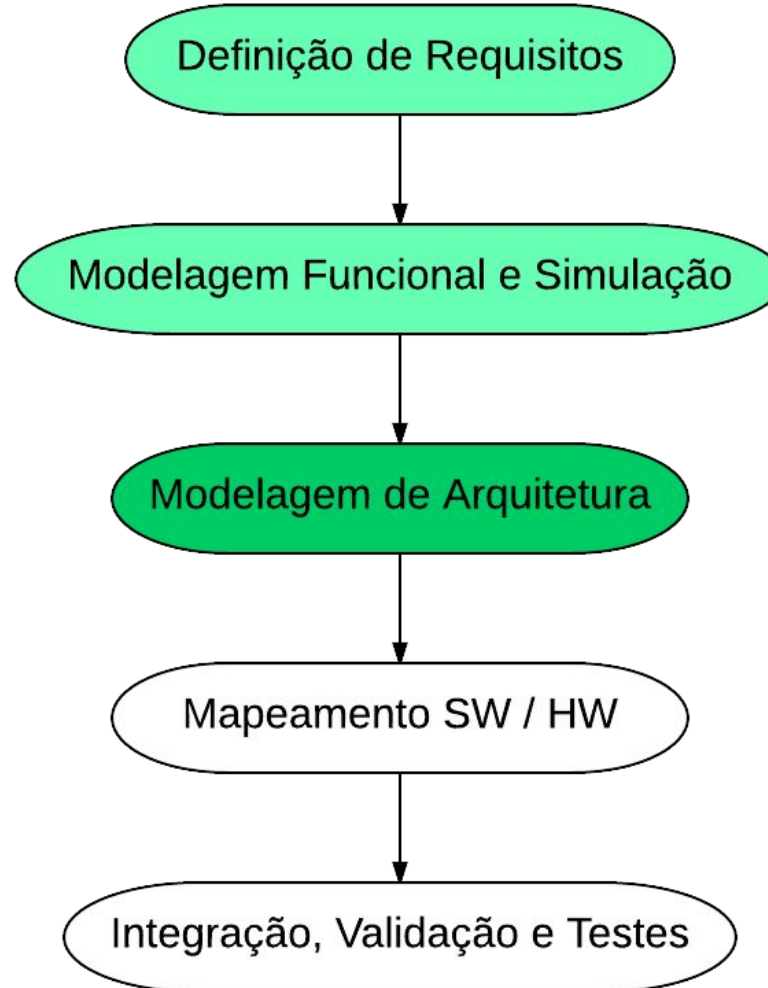
Modelagem Funcional - Diagrama de Blocos



Modelagem Funcional - Modelo a Eventos Discretos



Processo de Desenvolvimento de SEs



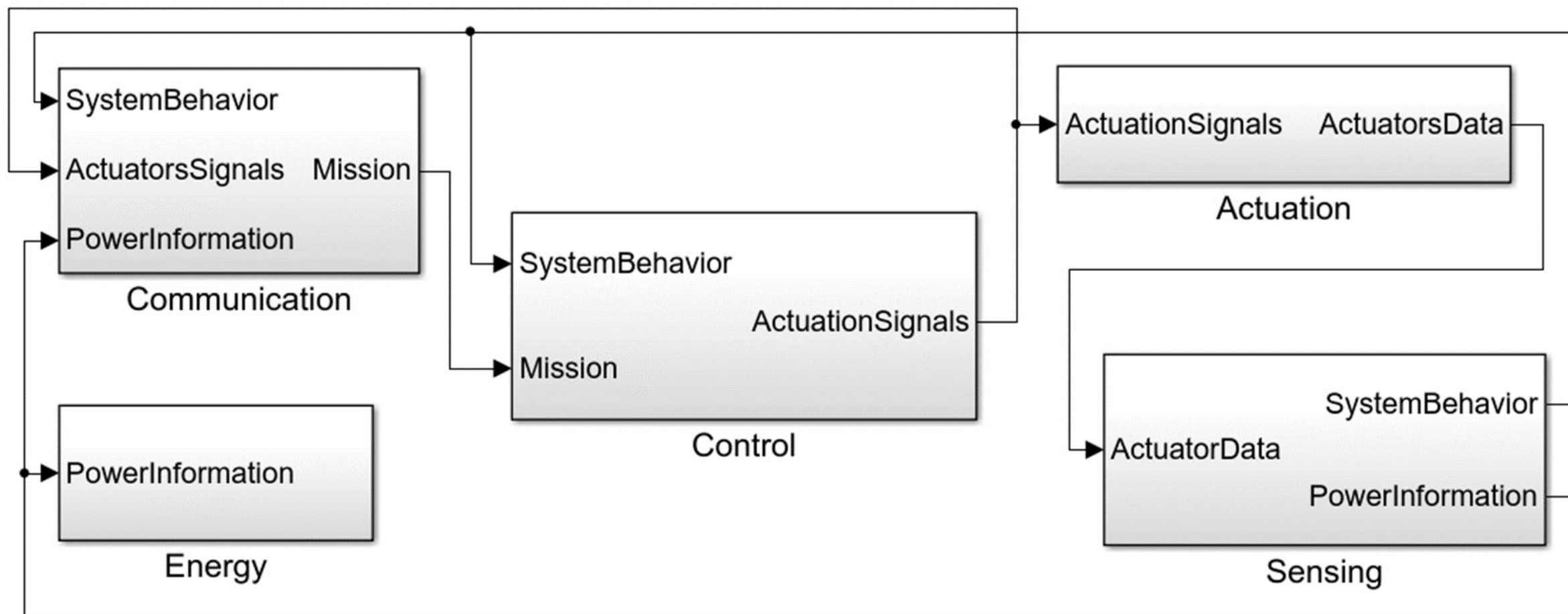
Modelagem de Arquitetura

- ☐ Define a estrutura ideal de HW / SW;
- ☐ Projeta as interfaces para componentes externos (Hardware, Software e Usuário);
- ☐ Definir entre processamento centralizado ou distribuído;
- ☐ Determinar concorrência entre tarefas;
- ☐ Estratégias de armazenamento de dados, manutenção e alocação de memória;

Modelagem de Arquitetura

- ☐ Projeto de algoritmos e funções de processamento de dados;
- ☐ Tratamento de erros;
- ☐ Análise de desempenho;
- ☐ Projeto detalhado dos componentes definidos na arquitetura do sistema;
- ☐ Documentação na forma de diagramas;

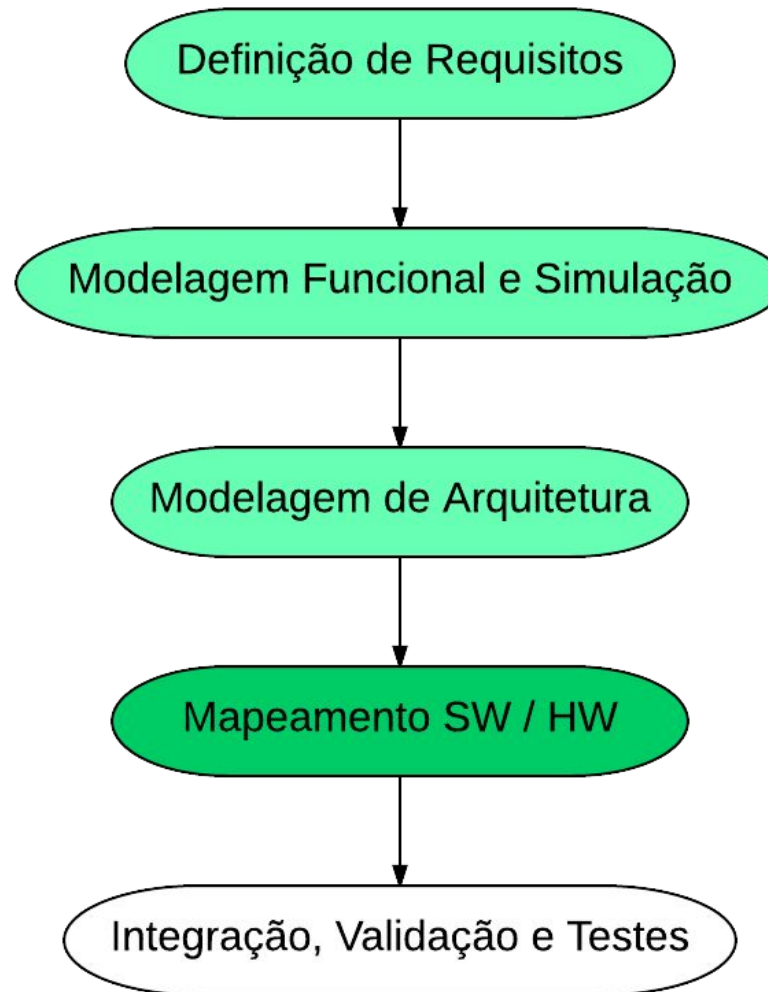
Modelagem de Arquitetura



Modelagem de Arquitetura

```
1 SYSTEM IMPLEMENTATION UAV.impl
2   SUBCOMPONENTS
3     --PROCESS
4     pi_energy: PROCESS p_energy.impl;
5     pi_control: PROCESS p_control.impl;
6     pi_sensing: PROCESS p_sensing.impl;
7     pi_actuation: PROCESS p_actuation.impl;
8     pi_communication: PROCESS p_communication.impl;
9     --DEVICE
10    di_gps: DEVICE d_gps.impl;
11    di_imu: DEVICE d_imu.impl;
12    di_esc_r: DEVICE d_esc.impl;
13    di_esc_l: DEVICE d_esc.impl;
14    di_sonar: DEVICE d_sonar.impl;
15    di_radio: DEVICE d_radio.impl;
16    di_servo_r: DEVICE d_servo.impl;
17    di_servo_l: DEVICE d_servo.impl;
18  CONNECTIONS
19    C1: PORT di_gps.position -> pi_sensing.position;
... Here goes all the connections (lines 20 to 45)
46 END UAV.impl;
```

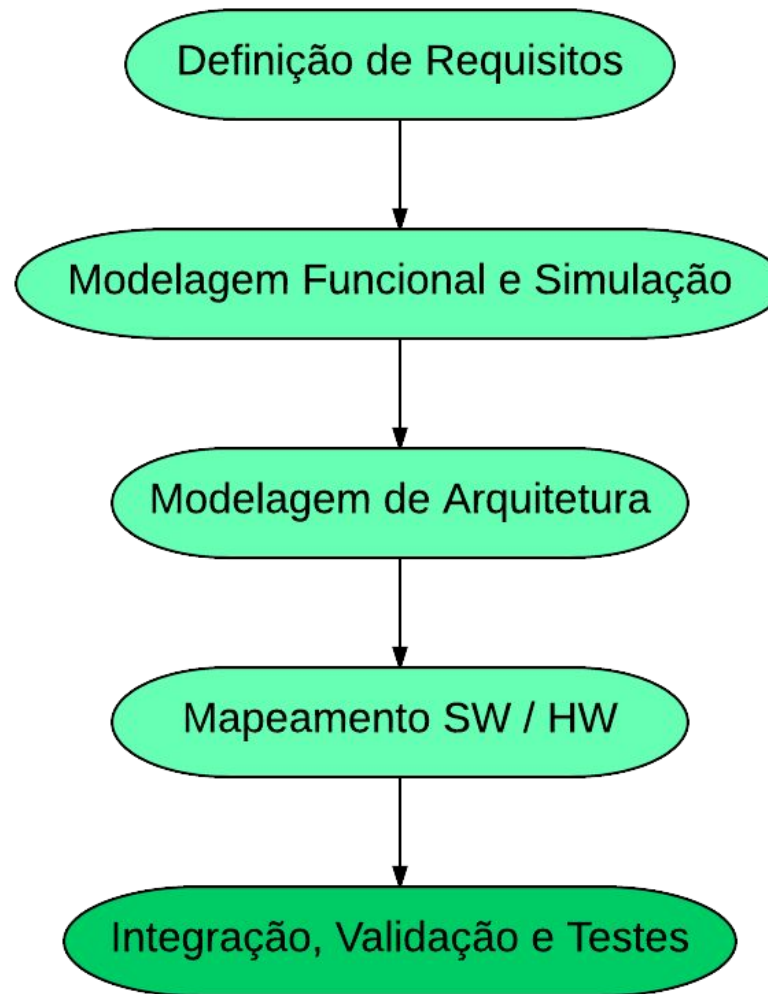
Processo de Desenvolvimento de SEs



Mapeamento SW / HW

- ☐ Deve atender aos requisitos Não Funcionais:
 - Restrições temporais;
 - Determinismo;
 - Custo;
 - Etc.
- ☐ Permite extrair informações sobre o tempo de execução (WCET);

Processo de Desenvolvimento de SEs



Integração, Validação e Testes

- ☐ Integração dos componentes do sistema na arquitetura final;
- ☐ Criação de casos de testes;
- ☐ Ajustes de parâmetros
- ☐ Refinamento dos modelos;
- ☐ Análises de latência;
- ☐ Verificação formal;

Conclusões

☐ Desafio tecnológico:

- Construir sistemas com qualidade, garantia de funcionalidade e custos aceitáveis;

☐ Desafio científico:

- Consolidar o desenvolvimento de SEs como uma disciplina científica e de engenharia;

☐ Necessita-se de ferramentas de apoio ao projeto de SEs.

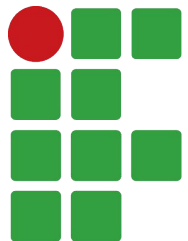
Conclusões

- ❑ Assim como sistemas computacionais convencionais, sistemas embarcados devem seguir um processo de desenvolvimento;
- ❑ A simulação das funcionalidades nos modelos de alto nível de abstração é importante;
- ❑ Cuidados especiais são necessários no projeto de arquitetura;
- ❑ Técnicas de Engenharia Baseada em Modelos auxiliam na transição entre diferentes visões do sistema.

Referências

- ❑ HEATH, S., Embedded Systems Design, 2 ed. Newnes, Oxford, 2003.
- ❑ BERGER, A. S., Embedded Systems Design, An Introduction to Processes, Tools, & Techniques, CMP Books, Berkeley, 2002.
- ❑ MARWEDEL, P., Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2 ed. Springer, New York, 2011.
- ❑ LEE, E. and SESHIA, S., Introduction to Embedded Systems, A Cyber-Physical Systems Approach, Berkeley, ISBN 978-0-557-70857-4, 2011.
- ❑ GANSSLE, J. G., The Art of Programming Embedded Systems, Academic Press, Califórnia, 1992.
- ❑ BECKER, L. B., FARINES, J., BODEVEIX, J., FILALI, M., e VERNADAT, F., Development process for critical embedded systems. Em Workshop de Sistemas Embarcados, Gramado. Anais. Porto Alegre: SBC, páginas 95{108, 2010.
- ❑ JENSEN, J. C., CHANG, D. H., e LEE, E. A., A model-based design methodology for cyber-physical systems. In Wireless Communications and Mobile Computing Conference (IWCMC), 2011.





INSTITUTO FEDERAL

Santa Catarina
Câmpus Tubarão

Obrigado!

Fernando Silvano Gonçalves

fernando.goncalves@ifsc.edu.br

se.cst.tub@ifsc.edu.br