



**ESCOLA POLITÉCNICA
UNIVERSIDADE DE SÃO PAULO**



**DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO
E SISTEMAS DIGITAIS
PCS**

PROJETO DE SISTEMAS EMBARCADOS

Prof. Dr. Carlos Eduardo Cugnasca

02/2018

Projeto de Sistemas Embarcados

Apresenta um sumário dos principais aspectos relacionados com Projetos de Sistemas Embarcados.

1. Introdução

Com o aumento da capacidade dos dispositivos eletrônicos e a redução de suas dimensões e custos, a Eletrônica Embarcada passou a ser utilizada em muitos produtos e equipamentos. Trata-se de uma solução eficaz quando comparada com as soluções convencionais, além de tornar as soluções cada vez mais poderosas e sofisticadas. Dessa forma, o termo Sistemas Embarcados passou a predominar, por espelhar as novas soluções de projeto. Esses sistemas incluem interfaces, dispositivos eletromecânicos, sensores e atuadores, e, naturalmente, a Eletrônica Embarcada.

A demanda crescente por novos Sistemas Embarcados exige que técnicas de projeto adequadas sejam adotadas, visando a geração de produtos que atendam aos requisitos estabelecidos.

2. Principais Características dos Sistemas Embarcados

Algumas características estão presentes na maior parte dos Sistemas Embarcados:

- **Aplicação Dedicada:** diferentemente de um computador, eles são projetados para realizar uma ou poucas funções, que não podem ser alteradas, mas apenas configuradas pelo usuário.
- **Interação com o Ambiente:** normalmente possuem alguns sensores e/ou atuadores que possibilitam a sua interação com o ambiente que o cerca.
- **Robustez:** eles devem resistir às condições de usos previstas em seu projeto e ao ambiente no qual atua. Uma falha pode ter grande impacto. Por exemplo, caso o a unidade de controle de um veículo deixe de funcionar, certamente o veículo terá que ser guinchado. Assim, os sistemas embarcados devem apresentar:
 - (i) boa **estabilidade** de funcionamento, recuperando-se automaticamente de algumas falhas que possam ocorrer;
 - (ii) alta **disponibilidade** (sistemas estáveis e que se recuperam costumam apresentar alta disponibilidade); e
 - (iii) **segurança**, tanto em relação à sua operação, minimizando os danos que possa alguma falha causar, como em relação às informações que manipula.
- **Eficiência garantida por Projeto:** uma série de características vem sendo comumente encontradas em projetos de sistemas embarcados:
 - **Baixo Consumo de Energia:** muitos sistemas são alimentados por baterias e devem possuir mobilidade, devendo consumir o mínimo possível.

- **Dimensões Reduzidas:** tanto a mobilidade como os locais de instalações em muitos casos exigem soluções de tamanho e peso reduzidos.
- **Baixo Custo:** por serem utilizados em produtos de mercados muito competitivos, soluções caras tendem a ser cada vez mais substituídas por outras mais econômicas.
- **Projeto Eficiente:** devido às restrições anteriores, tais sistemas se baseiam em processadores ou circuitos com capacidade restrita, tanto de memória, como de velocidade de processamento (quanto maior o *clock*, maior o consumo) e de tensão de alimentação. Procura-se sempre utilizar o menor número possível de componentes eletrônicos. O seu programa (também conhecido como *firmware*) deve respeitar as limitações de memória existentes e operar muitas vezes em tempo real.
- **Interfaces Simples:** devidos às restrições apresentadas, tais sistemas podem até mesmo não possuir interface com o usuário, e quando possuem, são simples (botões, leds, etc).

3. Projeto de Sistemas Embarcados

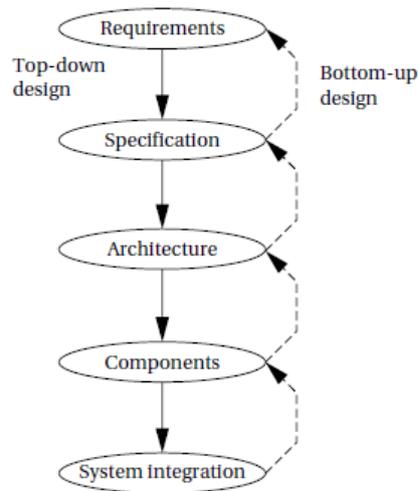
Para que os requisitos apresentados sejam atingidos, emprega-se alguma metodologia no projeto dos sistemas embarcados, que define os passos que devem ser seguidos no projeto, desde a sua concepção, passando pela implementação, teste e validação.

Uma boa metodologia possibilitará a geração da documentação técnica adequada do projeto, possibilitando a sua evolução futura (projetos com documentação deficiente tendem a ser descartados prematuramente). A boa metodologia possibilita o uso de **ferramentas de projeto** modernas, que reduzem o tempo e custos de desenvolvimento, além de possibilitarem projetos de melhor qualidade. Também metodologias que favoreçam a modularização e o trabalho colaborativo devem ser preferidas.

3.1 Exemplo de Metodologia

Waine Wolf, em seu livro [01], propõe, apresenta e discute metodologias de desenvolvimento de sistemas embarcados, conforme mostra a Figura 1.

Figura 1 – Etapas de uma Metodologia de Projetos



Fonte: extraído de [01]

Cinco etapas são propostas:

Etapa 1 - Requisitos

Envolve o levantamento do que se deseja projetar, a partir das informações do solicitante do projeto, e a sua descrição clara e sem ambiguidades.

Alguns consideram a fase mais importante para o sucesso, pois especificações errôneas e incompletas levam ao desenvolvimento de algo que não atenderá às expectativas. Deve-se, sempre que possível, identificar e eliminar as condicionantes fictícias, ou seja, imposições ou falsos requisitos fornecidos pelo solicitante, mas que na verdade não se sustentam, embora possam comprometer todo o projeto.

Os requisitos podem ser de duas naturezas:

- **Requisitos Funcionais:** relacionados com o funcionamento do sistema, indicando suas funções.
- **Requisitos Não-Funcionais:** indicam as propriedades do sistema, como desempenho, custo, tamanho, peso, e consumo de energia.

Etapa 2 - Especificação

Nessa etapa os requisitos são descritos formalmente, em uma linguagem sem ambiguidades, e validados pelo solicitante.

Etapa 3 - Arquitetura

O sistema é descrito funcionalmente, utilizando-se normalmente diagramas em blocos, destacando-se os principais componentes e módulos do sistema, bem como as suas conexões e funções. Após refinamentos sucessivos, os blocos são implementados por componentes.

Etapa 4 - Componentes

São utilizados dispositivos comercialmente disponíveis, e na ausência destes, os componentes são projetados e implementados. Tanto os componentes de software como os de componentes de hardware devem ser projetados de modo a respeitar as interfaces de conexão que foram padronizadas no projeto. O teste isolado dos componentes é recomendado, por simplificar a próxima etapa.

Etapa 5 - Integração do Sistema

Normalmente, nesta etapa são descobertos grande parte dos defeitos (*bugs*).

Dependendo da complexidade do projeto, a integração pode ser uma etapa de maior dificuldade, por envolver partes desenvolvidas por equipes distintas. A qualidade da metodologia e gerenciamento empregados, bem como das ferramentas de desenvolvimento empregadas, pode minimizar tais dificuldades.

3.2 Abordagens para o Desenvolvimento de um Projeto

A literatura destaca duas principais abordagens que podem ser adotadas no desenvolvimento de sistemas:

- **Bottom-up:** as partes ou módulos do sistema são projetados e testados separadamente, antes de serem combinados para constituir um sistema maior. O projetista parte do nível mais detalhado para o mais abstrato, expandindo e modificando os componentes, de modo a atender aos requisitos do sistema.
- **Top-down:** é a mais popular, utilizada para o projeto de muitos tipos de sistemas. Utiliza-se o processo de decomposição e refinamento sucessivo para que o objetivo final seja atingido. Tenta-se identificar todas as funções do sistema, que são particionadas em sub-funções menos complexas, cada uma desempenhando uma tarefa específica. Sucessivamente as sub-funções também são subdivididas em tarefas menos complexas, até que elas apresentem baixa complexidade e, portanto, são de fácil implementação e teste. Assim, pode-se reduzir a complexidade do software e do hardware de um sistema a proporções gerenciáveis.

Muitos especialistas consideram a metodologia *top-down* mais lógica e sistemática do que a *bottom-up*. A cada subdivisão das funções, maior quantidade de detalhe é acrescentada. Parte-se de um nível conceitual, observando-se a estrutura global e ampla do sistema, antes do acréscimo de mais detalhes. Essa metodologia favorece também a manutenção de um nível de qualidade constante, através de pontos de avaliação ao longo de todo o projeto. Trata-se de uma metodologia fundamental para o projeto de software e de sistemas como um todo.

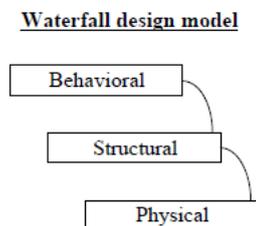
A metodologia *bottom-up* é mais útil no projeto do hardware, que em geral consiste no uso de componentes com alto grau de integração como circuitos integrados ou placas disponíveis no mercado.

O processo de desenvolvimento desses sistemas visa obedecer a uma dada especificação a custos e desempenho competitivos. Isso requer a organização das diversas partes do sistema de forma coerente e eficiente. Para tal, é necessário a tradução das especificações do sistema a ser implementado em uma arquitetura constituída por unidades funcionais de tecnologia eletrônica.

Existem outras abordagens que vêm ganhando destaque:

- **Cachoeira (Waterfall):** segue a abordagem *top-down*, e cada fase do projeto é executada uma vez, sem que haja *feedback* da fase seguinte para a fase anterior. Por essa razão, ela é menos utilizada, pois em geral os desenvolvedores realizam testes em uma fase inferior e voltam à fase anterior, alterando-a em função dos testes.

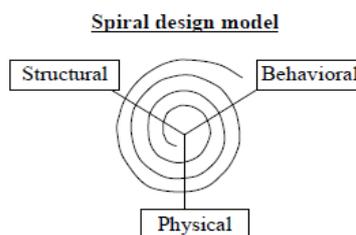
Figura 2 – Modelo de Projeto Waterfall



Fonte: extraído de [01]

- **Espiral:** prevê o detalhamento sucessivo do projeto, por meio de um protótipo simplificado ou simulação, que proporciona o aprendizado adquirido a cada inclusão de novas funcionalidades e testes, que possibilita a descoberta de requisitos não previstos originalmente. Um ponto desfavorável dessa abordagem é o alongamento do projeto, caso muitos detalhamentos sejam realizados.

Figura 3 – Modelo de Projeto Espiral



Fonte: extraído de [01]

- **Refinamentos sucessivos:** interessante quando a especificação do problema não é detalhada nem clara devido ao desconhecimento que se deseja realmente do projeto a ser desenvolvido. São realizados refinamentos do sistema até a clareza ser atingida. A partir de um projeto inicial simples constrói-se um protótipo que gradativamente recebe novos recursos, a partir dos destes realizados, até que o projeto a ser desenvolvido esteja adequadamente especificado.

3.3 Uma Metodologia de Projeto de Sistemas Embarcados [02]

3.3.1. Introdução

O elevado estágio de qualidade e sofisticação das tecnologias de hardware e de software atualmente não é, por si só, garantia de satisfação tanto dos usuários dos sistemas, que desejam confiabilidade e funcionalidade, como dos que os desenvolvem e os fornecem, que enfrentam longos processos de depuração, a altos custos [03]. O aproveitamento efetivo das potencialidades da tecnologia, e também dos recursos humanos, é muito dependente da metodologia de desenvolvimento selecionada.

Um ponto que deve ser destacado a partir do que foi apresentado é que a constante interação de questões abstratas, como a especificação de um projeto, com as concretas, como o protótipo de um projeto. Essa constante interação pode ser formalizada, incorporando aspectos das abordagens Espiral e de Refinamentos Sucessivos.

A seguir, é apresentado um modelo de desenvolvimento de sistemas, que pode ser empregado no desenvolvimento de sistemas embarcados [03]. Ela é sustentada por dois fundamentos:

- a interação entre a **realidade** e o **pensamento**;
- a interação entre o **problema** e a **solução**.

Dois planos fundamentais são caracterizados, como mostra a Figura 4.

Figura 4 - Interações entre a realidade e o pensamento, e entre o problema e a solução



Fonte: baseada em [04]

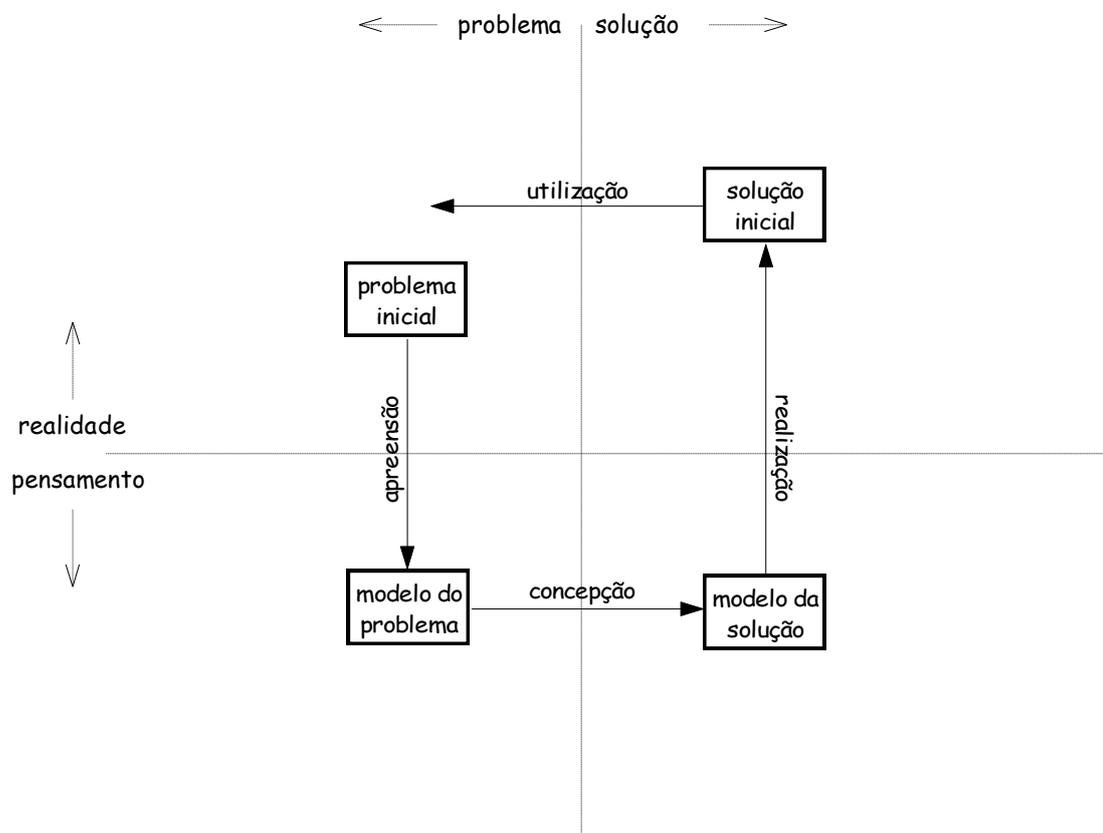
- o plano da **realidade**, correspondendo ao mundo físico, constituído por entidades não transformáveis pela intervenção do homem - o espaço e o tempo, e por entidades que podem ser transformadas pela intervenção do homem - os objetos e as ações;
- -o plano do **pensamento**, correspondendo ao mundo abstrato, constituído por noções de espaço e de tempo, e por ações e objetos abstratos.

Nesses planos, têm-se dois conceitos associados:

- o **problema**, correspondendo à impossibilidade de uma ação desejada pelo homem, ou à sua pouca eficiência;
- a **solução**, associada à superação de impossibilidades ou melhorias na realidade existente através de uma ação.

A metodologia de desenvolvimento de sistemas em questão corresponde a interações entre o problema e a solução, nos planos da realidade e do pensamento. Definem-se quatro tipos de ações entre os elementos do modelo (Figura 5).

Figura 5 - As atividades de um projeto de Engenharia

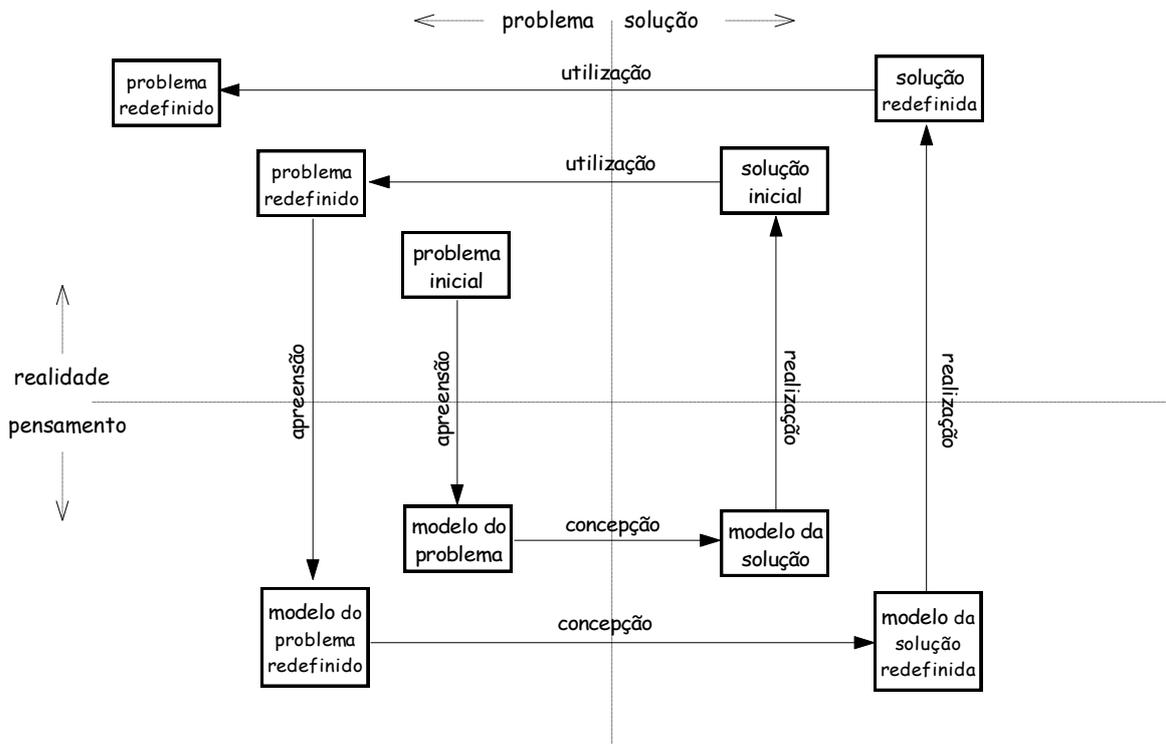


Fonte: baseada em [04]

- a **apreensão**, ou forma de construção de uma representação abstrata para um problema real;
- a **concepção**, ou criação da solução para um problema no plano do pensamento;
- a **realização**, ou a construção de uma solução no plano da realidade;
- a **utilização**, ou a instauração da solução no ambiente do problema.

O exercício dessas ações como procedimento de resolução de um problema inicial (também denominado de demanda inicial), poderá acarretar transformações na realidade, dando origem a novas transformações, a partir de uma redefinição do problema. Tem-se, pois, uma espiral evolucionária, que contribui com sucessivos aprimoramentos na solução, até que esta atinja um índice de qualidade satisfatório (Figura 6).

Figura 6 - O aspecto evolucionário do desenvolvimento de projetos de Engenharia



Fonte: baseada em [04]

3.3.2. A Aplicação da Metodologia

Na aplicação da metodologia apresentada para o desenvolvimento de um sistema embarcado podem-se identificar basicamente duas grandes fases: a sua **especificação** e a sua **implementação**.

A especificação de um sistema se caracteriza pela obtenção de uma solução para o usuário, com auxílio da metodologia em questão. Contudo, ela também corresponde a um problema inicial para quem vai desenvolver ou fornecer o sistema, devendo o mesmo ser tratado como sugere a metodologia.

Outro aspecto comum de ser identificado na fase de implementação é a decomposição do problema original em problemas menores e, portanto, menos complexos, e individualmente de maior simplicidade de implementação. Pode-se, assim, proporcionar sucessivas divisões do sistema a ser implementado em termos de subsistemas, até a obtenção de modelos de maior facilidade de implementação.

Em desenvolvimentos de sistemas de pequeno e médio porte, é frequente a adoção de um ciclo de vida dinâmico de desenvolvimento [03][05], envolvendo a prototipação. Nesses casos, pressupõe-se que:

"O conjunto completo de especificações essenciais de um sistema não será descoberto até que o usuário tenha a oportunidade de experimentá-lo." [06]

Dentro dessa abordagem, tem-se que após sucessivas interações e modificações no protótipo de um sistema, obtêm-se os resultados desejados. Contudo, para que esse modelo seja eficiente, devem-se criar condições para que o tempo consumido a cada interação não seja demasiadamente grande.

Essa filosofia de desenvolvimento fundamentalmente reflete o espírito da engenharia, que procura aplicar os conhecimentos científicos e os procedimentos empíricos à criação de equipamentos, sistemas, métodos e processos.

Em sistemas de maior porte, costuma-se adotar o ciclo de vida clássico [07], no qual a especificação de requisitos é feita pelo usuário, de forma muito precisa, somente após o que se dará origem à implementação.

Concluindo-se, tem-se que, nos dias atuais, é indiscutível a necessidade de métodos adequados ao desenvolvimento de sistemas embarcados. E uma vez que os métodos nada mais são do que regras originadas de uma atenta e crítica observação dos procedimentos humanos para a obtenção da solução de algum problema torna-se por demais proveitosa a explicitação do método adotado em cada caso, para que o mesmo possa ser avaliado e aprimorado ao longo de todo o processo de desenvolvimento, e orientar novos trabalhos na área.

4. Alternativas de Implementação

Um projeto típico de um sistema embarcado envolve basicamente o desenvolvimento do seu software e do seu hardware. Ambos podem aproveitar componentes já desenvolvidos e testados (circuitos integrados, placas, bibliotecas de software, etc). Dependendo do porte do trabalho, cada parte poderá estar a cargo de uma equipe distinta, o que exigirá uma tarefa de gerenciamento do projeto, para uma atuação harmoniosa entre as partes envolvidas.

Cada vez mais estão disponíveis módulos eletrônicos que podem ser integrados de modo a constituir um produto ou protótipo inicial desse produto, com rapidez e baixo custo. O mesmo pode-se dizer do software, encontrando-se bibliotecas prontas e testadas para uma grande variedade de necessidades. O trabalho do projetista deve considerar o aproveitamento das partes de qualidade encontradas no mercado, pois um projeto completo de um sistema, como se fazia no passado, será demorado e custoso.

Mesmo que a integração de módulos pode seguir diversas abordagens, sendo as principais descritas a seguir.

4.1. Projeto baseado em Processadores de Propósito Geral

O microprocessador de propósito geral possibilita soluções complexas e poderosas, porém mais custosas, além de maiores e de consumirem mais energia. São flexíveis e podem fazer uso de paralelismo, obtendo-se maior desempenho do que outras soluções. Porém, nem sempre são mais adequados para aplicações em tempo real, devido ao compartilhamento de recursos e paralelismo, tornando o seu processamento menos determinístico. Seu uso se justifica em aplicações que necessitam alto desempenho, e o custo e consumo não são tão críticos.

4.2. Projeto baseado em Microcontroladores

O microcontrolador é naturalmente a primeira opção de um projeto de sistemas embarcados, por possibilitar implementações com poucos componentes, e que podem ser realizadas com bons recursos de desenvolvimento (montadores, compiladores, simuladores, bibliotecas, etc). Sistemas embarcados necessitam de um processador a partir do qual as interfaces com sensores, atuadores, usuários e outros dispositivos são implementadas. Projetos com microcontroladores são desenvolvidos desde a década de 1970 sendo, portanto, uma tecnologia consolidada e ensinada nas escolas de engenharia.

Possibilitam escalamento e sincronização de processos, interrupções, e apresentam baixa latência, sendo adequados para aplicações em tempo real.

Os microcontroladores possuem internamente os principais recursos requeridos por sistemas de pequeno e médio porte, como entradas e saídas digitais e analógicas, canais de comunicação e memória. Possibilitam soluções compactas de baixo consumo e baixo custo.

Cabe ao projetista selecionar o mais adequado a cada aplicação, em função dos requisitos. Essa tarefa naturalmente exige alguma experiência.

4.3. Projeto baseado em Hardware Programável (FPGA)

Uma FPGA (Field Programmable Gate Array) é um circuito integrado projetado para ser configurado pelo usuário, e que possibilita a implementação de um hardware reconfigurável. Para a descrição do hardware a ser configurado utilizam-se linguagens especiais, como VHDL, utilizada para descrever o comportamento e estrutura de um sistema digital. VHDL significa VHSIC Hardware Description Language, e VHSIC significa Very High Speed Integrated Circuit. Por meio dela é possível descrever e simular muitos tipos de circuitos digitais. Ela pode descrever o comportamento, o fluxo de dados e a estrutura do circuito. Os projetos são baseados na metodologia *top-down*, sendo que os sistemas são especificados e testados em alto nível, permitindo refinamento após ter sido depurado.

Como vantagens do seu emprego, tem-se a facilidade de simulação e teste, sendo uma boa alternativa para sistemas críticos. Porém pode ser uma alternativa custosa para pequenos projetos, como muitos sistemas embarcados.

4.4. Projeto baseado em Application-Specific Integrated Circuits (ASICs)

Os circuitos integrados de aplicação específica, ou Application Specific Integrated Circuits (ASICs) destinam-se à implementação de requisitos específicos de um projeto, podendo apenas ser utilizado para essa finalidade. Linguagens de descrição de hardware (HDL), como Verilog ou VHDL, são utilizadas para descrever as funcionalidades do ASICs. Possibilita projetos otimizados (System-on-chip – SoC), de bom desempenho e baixo consumo de energia. Contudo, os projetos são inflexíveis, requerem maior tempo para serem projetados e construídos, e apresentam alto custo. Assim, o seu uso é viabilizado para produtos que são comercializados em larga escala, como eletrodomésticos, smartphones, etc.

4.5. Projeto baseado em Plataformas

O projeto baseado em plataformas é utilizado há muito tempo, desde a época dos PCs. Atualmente estão disponíveis plataformas que podem ser utilizadas para o desenvolvimento de sistemas embarcados, em especial protótipos ou sistemas que inicialmente não se pretende comercializar em grande escala.

Placas como Arduino, Raspberry, são opções interessantes para prototipação, por apresentarem baixo custo, muitos módulos sensores e atuadores compatíveis, ferramentas de suporte ao desenvolvimento, farta literatura e comunidades que fornecem suporte. Muitos trabalhos acadêmicos, de pesquisa e de hobbistas são implementados com esses dispositivos.

Após a validação de um protótipo, pode-se simplificar o projeto de forma mais otimizada e econômica, projetando-se placas dedicadas baseadas na implementação do protótipo.

Referências

- [01] WOLF, W. **Computers as Components - Principles of Embedded Computing System Design**. Morgan Kaufmann Publishers, San Francisco, 2ed., 2008.
- [02] CUGNASCA, C. E. **Técnicas de Desenvolvimento de Projetos Baseados em Microprocessadores**. Apostila da disciplina PCS 2497 Laboratório de Processadores I, Departamento de Engenharia de Computação e Sistemas Digitais. Escola Politécnica da USP, 2006.
- [03] SOARES, J. O. P. **Especificação de Métodos de Desenvolvimento de Sistemas - Aplicação a Sistemas de Tempo Real**. São Paulo, 1986. Dissertação (Mestrado) - Escola Politécnica, Universidade de São Paulo.
- [04] SOARES, J.O.P. **Fundamentos conceituais para projeto de aplicações de computadores**. São Paulo - SP, 1993. Tese (Doutorado) – Escola Politécnica da Universidade de São Paulo.
- [05] BLUM, B.I. The Life Cycle - a Debate over Alternate Models. **ACM SIGSOFT Software Engineering Notes**, 7 (4): 18-9, Oct., 1982.
- [06] CONNELL, J. & BRICE, L. Rapid Prototyping. **Datamation**, p. 93-100, ago., 1984.
- [07] BOEHM, B.W. Software Engineering. **IEEE Transaction on Computer**. C25 (12): 1226-41, Dec., 1976.