

Atividades 5 e 6

Kit de desenvolvimento e C para embarcados

Rafael Corsi
rafael.corsi@insper.edu.br

6 de outubro de 2016



Entregar na 3ª aula impresso e em formato PDF via github.

1 Overview

Quando trabalhamos computação embarcada, devemos ter um conhecimento mais profundo do alvo do programa (hardware) já que não trabalhamos com uma camada de abstração de hardware tal como um sistema operacional de alto nível (Linux, Windows, Androide, ...).

O correto entendimento do sistema é crucial para o bom funcionamento do programa, já que iremos trabalhar diretamente com o acesso de entradas e saídas reais (I/Os).

Podemos classificar esse entendimento em 3 níveis :

- Microcontrolador
- Hardware (Placa de circuito Impresso - PCB)
- Programa (firmware)

O microcontrolador é o dispositivo central do hardware, será o responsável pelo controle e supervisão dos demais componentes da placa.

O hardware é um conjunto de componentes eletrônicos (microcontrolador inclusive) que forma uma eletrônica embarcada. No hardware estão partes como : fonte de alimentação, memórias externas, conectores, LCD.

Questão. 1.1: Diagrama de blocos

Esboce um diagrama de blocos que ilustre a interação entre o uC, Hardware e firmware.

! Dica: Existem dois documentos que serão muito utilizados ao longo do curso: *Cortex-M7-SAM-E70* e *SAM-E70-XPLD*.
O primeiro é o datasheet do microcontrolador utilizado no kit de desenvolvimento e o segundo é o manual que explica como o microcontrolador é utilizado no kit.

2 SAM-E70 microcontrolador

(Utilize o manual encontrado em : *Kit de Desenvolvimento/SAM-E70-XPLD* para resolução dessa secção.)

Com um mercado de 16.5 bilhões de dólares em 2013 ¹ existe uma enorme diversidade de microcontroladores, cada um com uma especificidade. Conhecer o alvo do seu código é um fator primordial para o funcionamento da aplicação.

Questão. 2.1: Microcontrolador

Identifique e liste as especificidades do microcontrolador utilizado no curso.

Os microcontroladores ofertados pelo mercado diferenciam-se principalmente pela sua arquitetura (CORE), quantidade de memória interna (RAM, ROM), e pelos periféricos encontrados dentro do microcontrolador.

Questão. 2.2: Memória

Liste os tipos de memórias internas do microcontrolador SAM-E70 e seus tamanhos.

Questão. 2.3: Memória II

Porque é importante saber quanto de memória um sistema possui ?

2.1 Periféricos

Periféricos são pequenos dispositivos eletrônicos que junto com o CORE mais memórias compõem um microcontrolador. Os periféricos são responsáveis por executar tarefas simples simultaneamente ao processamento principal do CORE, muitas vezes, adicionam funcionalidades extras ao uC.

¹<http://www.grandviewresearch.com/industry-analysis/microcontroller-market>

A boa utilização dos periféricos torna o código mais robusto e permite a otimização da utilização do processamento do microcontrolador. Fig. 1 ilustra a arquitetura interna do microcontrolador SAM-E70, e seus periféricos.

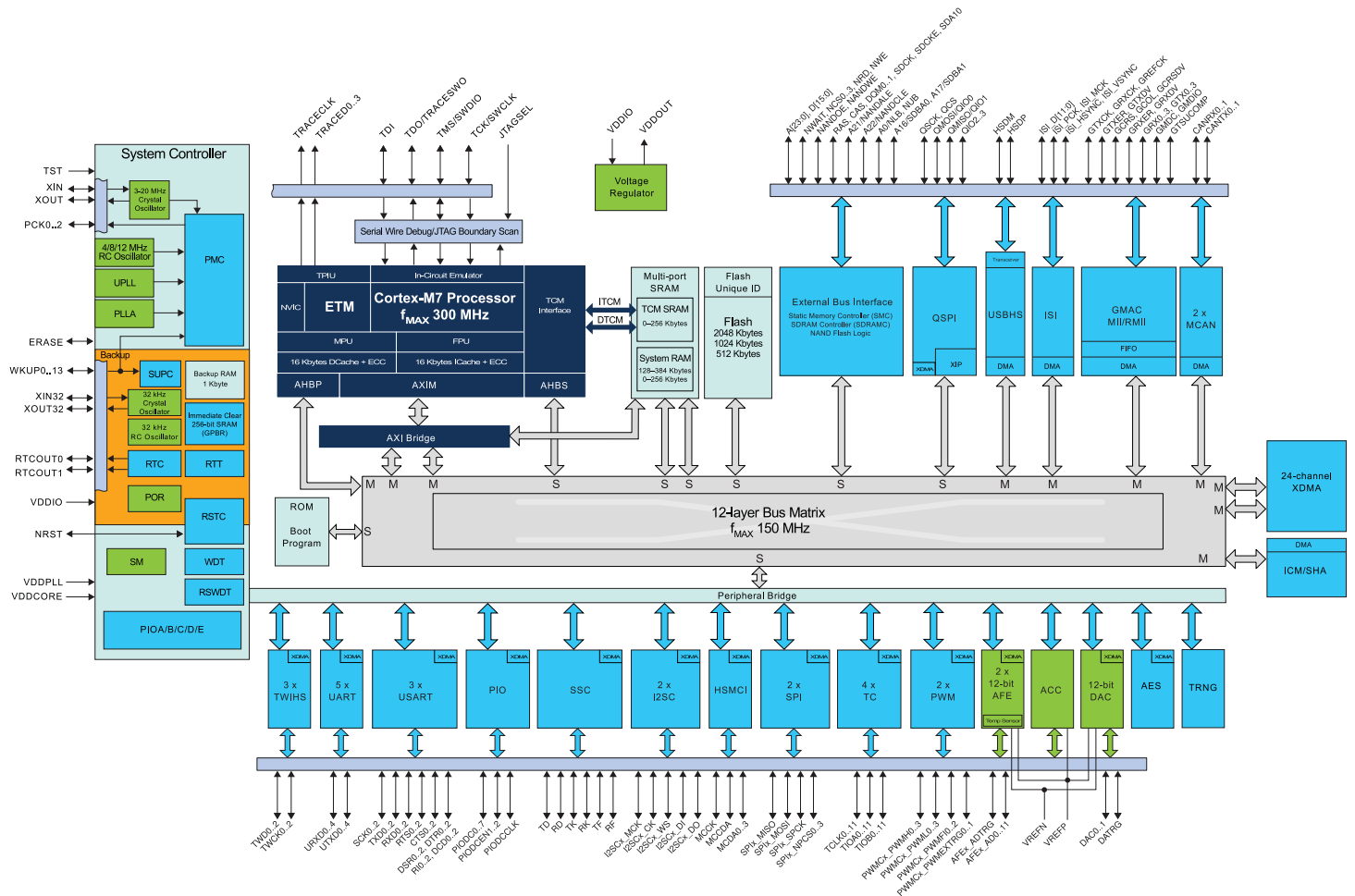


Figura 1: Diagrama de blocos SAM-E70
(Cortex-M7-SAM-E70 - pg. 7)

Questão. 2.4: Periféricos

Escolha um dos periféricos do microcontrolador (ADC, DAC, TC, USB, Ethernet) e explique sua funcionalidade.

2.2 Extras

Questão. 2.5: Watchdog

O que é watchdog timer e qual é sua utilização ?

Questão. 2.6: Custo

Pesquise nos fornecedores qual o valor de mercado do chip utilizado no kit de desenvolvimento SAM-E70.

3 SAM-E70-XPLD hardware

(Utilize o manual encontrado em : *Kit de Desenvolvimento/Cortex-M7-SAM-E70* para resolução dessa secção.)

Utilizaremos ao longo do curso um kit de desenvolvimento (hardware) desenvolvida pela Atmel chamada de SAM-E70 XPLD, esse kit possui em seu núcleo um processador ARM Cortex-M7, e algumas outras funcionalidades tais como: memória externa, SD-Card, Ethernet, USB. Essas funcionalidades extras são também chamadas de periféricos, mas agora não sendo referido aos periféricos do uC mas sim aos periféricos da placa.

A Fig. a seguir, extraída do manual do kit de desenvolvimento ilustra como os demais periféricos estão conectados no microcontrolador.

! Com os conceitos desenvolvidos até aqui, repense o diagrama da Questão 1.1

3.1 Gravador/Debug

Diferente da programação de um software em um computador, na computação embarcada é necessário utilizar ferramentas especiais para passarmos o binário compilado no PC para o alvo (uC).

Cada fabricante utiliza uma tecnologia de gravação específica, porém recentemente um formato conhecido como JTAG (*Joint Test Action Group*) tende padronizar a gravação dos uC.

Questão. 3.1: JTAG

Descreva como funciona a gravação via JTAG.

3.2 Clock

Todo sistema digital utiliza de um sinal chamado de *clock* para sincronização interna de toda a eletrônica, alguns dispositivos mais complexos como um microcontrolador, podem possuir diversos domínios de clock interno.



Figura 2: Periféricos SAM-E70-XPLD
(SAM-E70-XPLD - pg. 3)

Questão. 3.2: Clock

Qual a relação do clock no consumo de energia em sistemas eletrônicos ?

4 Firmware - Especificidades

! Firmware é o termo utilizado para descrever um software que executa em um microcontrolador, esse termo é utilizado para evitar confusões na descrição de um projeto.

Questão. 4.1: volatile/const/static

O que são variáveis volatile/const/static ?

Questão. 4.2: MakeFile

O que é um makefile e qual a sua utilização ?

Questão. 4.3: ASCII

O que é ASCII, e quando é utilizado ?

5 Primeiro projeto

Siga o tutorial em :

<https://github.com/insper/computacao-embarcada/wiki/Getting-Startted-Atmel-Studio-7>

Salve o projeto criado dentro da pasta do seu repositório.