

Software Embarcado

Francisco Sant'Anna
Sala 6020-B

`francisco@ime.uerj.br`

`http://github.com/fsantanna-uerj/SE`

Embedded software

From Wikipedia, the free encyclopedia

Embedded software is computer [software](#), written to control machines or devices that are not typically thought of as computers. It is typically specialized for the particular [hardware](#) that it runs on and has time and memory constraints.^[1]

Conteúdo

- Arduino
- GPIO / Entrada e Saída
 - Digital e Analógica (saída)
- Sensores e Atuadores
- Conversor analógico digital
- Temporizadores
- Comunicação Serial
 - USART, I2C, SPI
- Comunicação via Rádio
- Tratamento de Interrupções

Conteúdo

- É um curso **prático** de **Software**
- Muita programação
- Projetos

Segunda, M5M6

Quarta, M5M6

Sala 6023/2-F

Online

- GitHub

- <https://github.com/fsantanna-uerj/SE/>

- Grupo de e-mail

- <https://groups.google.com/d/forum/se-uerj/>

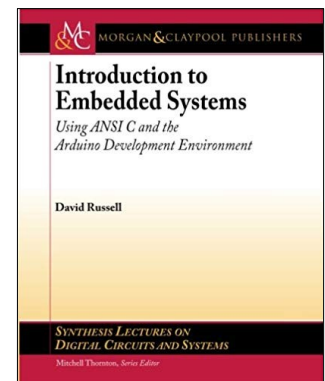
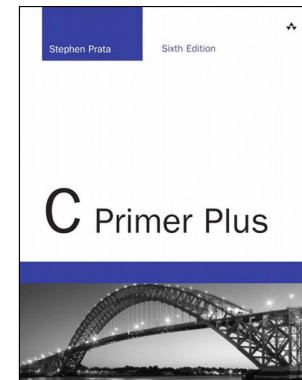
Email delivery preference: Notify me for every new message (fewer than 1 per day) ▼

Bibliografia

- *C - Primer Plus*
- *Introduction to Embedded Systems: Using ANSI C and the Arduino Development Environment*
- Vídeos/Curso do Robert Paz

- <https://www.youtube.com/watch?v=9Q-3c0gQcok&list=PLifLftIJFUm-1iIAEPWvuSJTA50YKYD7J>

- Todos em inglês
 - Mas não são essenciais



Aprovação

- 75% de presença
- Tarefas
- Projeto
- Prova (?)

Tarefa-00

(até domingo 11/08)

- Cadastrar-se no grupo da turma
 - <https://groups.google.com/d/forum/se-uerj/>
- Criar um repositório com o nome “SE” no GitHub
 - <https://github.com/>
 - Adicionar um arquivo README .md
 - texto “pessoal” qualquer formatado em *Markdown*
 - <https://help.github.com/articles/basic-writing-and-formatting-syntax/>
 - Mandar um e-mail para o grupo com o link do seu repositório

Tarefa-01 - Pisca

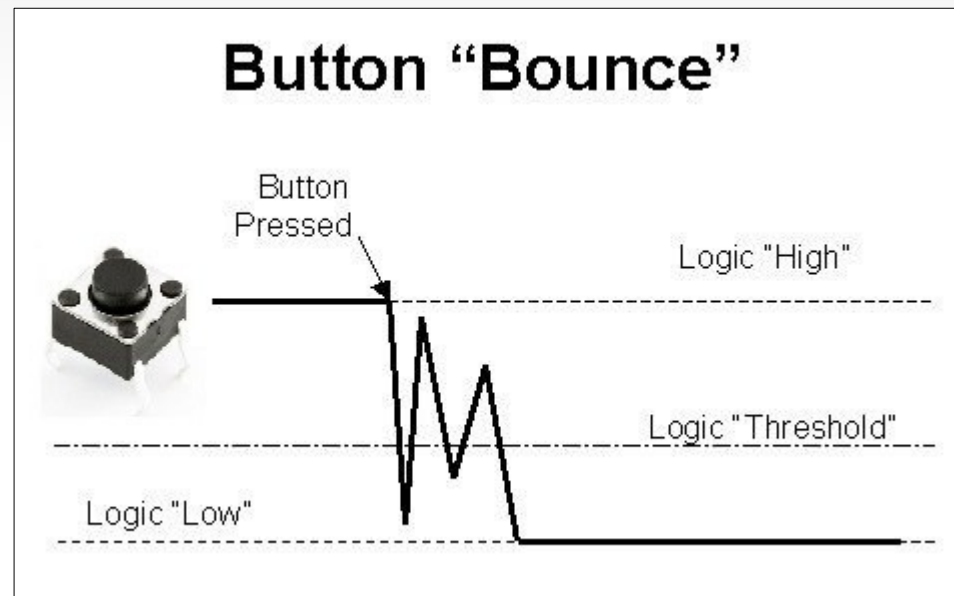
(até domingo 18/08)

- Piscar o LED a cada 1 segundo
- Botão 1: Acelerar o pisca-pisca a cada pressionamento
(somente ao soltar e não ao pressionar)
- Botão 2: Desacelerar a cada pressionamento
(somente ao soltar e não ao pressionar)
- Botão 1+2 (em menos de 500ms): Parar

Tarefa-01 - Debouncing

(até domingo 18/08)

- *Debouncing*



Tarefa-01 - FSM

(até domingo 25/08)

- Desenhar uma máquina de estados finita (FSM) para modelar o problema
- Modificar a implementação para refletir a modelagem

Tarefa-02 - Sensores e Atuadores

(Dom, 08/09)

- Explorar os sensores e atuadores do Kit
- Identificar cada um deles
 - Utilidade (pra que serve)
 - Circuito (como conectar)
 - API (programa de teste)
- Fazer uma aplicação qualquer usando todos eles
- Desenhar e usar a modelagem de FSMs
- Fazer uma apresentação de 5 minutos
 - Dica: usar o *Fritzing* para os circuitos

Tarefa-03 - Entrada Analógica

(Dom, 22/09)

- Usar o conversor ou comparador analógico
- [2020/1] duas entradas concorrentemente

Tarefa-04 - Temporizadores

(Dom, 29/09)

- Usar um timer para contar o tempo na Tarefa 2
 - Em modo Normal ou CTC (*clear timer on compare*)
 - Verificar flags (OVF ou OCF)
- Usar outro timer para uma das seguintes funções:
 - Capturar o tempo de uma entrada
 - Gerar uma forma de onda
- Usar o prescaler
- F_CPU guarda a frequência da CPU

Tarefa-05 - Pré-Projeto

(Qua, 09/10)

- Pré-Projeto Escrito:
 - projeto/README.md
 - descrição
 - componentes
 - **comunicação RF**
 - máquina de estados abstrata

Tarefas

- Tarefa-00: **Dom, 11/08**: Lista & GitHub
- Tarefa-01: **Dom, 18/08**: Pisca
- Tarefa-01: **Dom, 25/08**: Pisca - FSM
- Tarefa-01: **Dom, 01/09**: Pisca - FSM - Portas
- Tarefa-02: **Seg, 09/09**: Sensores e Atuadores
- Tarefa-03: **Dom, 22/09**: Conv. ou Comp. Analógico
- Tarefa-04: **Dom, 29/09**: Timers
- Tarefa-05: **Qua, 09/10**: Pré-Projeto Escrito

Próximas Aulas

22/10

A

24/10

A

29/10

V

31/10

V

05/11

V

07/11

V

12/11

A

14/11

A