

Lectures Notes on Embedded Systems

Camilo de Lellis

October 21, 2025

Contents

1	Aula - 11/09/2025	2
1.1	Ementa da Disciplina	2
1.2	Introdução aos sistemas embarcados	3
1.3	Introdução a microcontroladores	4
1.4	Conceitos e fundamentos	4
2	Aula - 15/09/2025	4
2.1	Entradas Digitais	4
3	Aula - 18/09/2025	4
3.1	Introdução à programação de microcontroladores em linguagem C	4
3.1.1	Tipos de dados	4
3.1.2	Tabela ASCII e Sistema Binário	4
3.1.3	Configuração de GPIO	4
4	Aula - 22/09/2025	5
4.1	Manipulando saídas digitais	5
4.1.1	Acionamento de LED	5
4.1.2	Corrente máxima nos pinos	5
4.1.3	Cálculo da resistência limitadora	5
4.1.4	Prática rápida	5
5	Aula - 25/09/2025	5
5.1	Programação de microcontroladores: Temporização	5
5.1.1	Função delay()	5
5.1.2	Função millis()	5
5.1.3	Espera ocupada	5
6	Aula - 29/09/2025	6
6.1	Prática sobre acionamento de cargas utilizando transistores	6
7	Aula - 02/10/2025	6
7.1	Programação com temporização: função millis()	6
8	Aula - 06/10/2025	6
8.1	Exercícios	6
8.1.1	Entradas digitais	6
8.1.2	Saídas digitais	6
8.1.3	Resistores de pull-up	6
8.1.4	Resistores de pull-down	6
8.1.5	Projeto de sistemas digitais utilizando arduino	6
8.1.6	Utilização dos operadores lógicos da linguagem C	6
9	Aula - 09/10/2025	6
9.1	Programação com temporização: função millis(). Aplicações	7

10 Aula - 13/10/2025	7
10.1 Acionamento de cargas utilizando relé eletromecânicos	7
10.1.1 Princípio de funcionamento	7
10.1.2 Simbologia	7
10.1.3 Exemplo de aplicação	7
11 Atividade - 17/10/2025	7
11.1 Código em C++	7
12 Simulado da 1ª avaliação - 20/10/2025	8
13 Atividade: Genius da Estrela - 23/10/2025	8

1 Aula - 11/09/2025

Conteúdos ministrados: Apresentação da ementa. Introdução aos sistemas embarcados, microcontroladores, conceitos e fundamentos.

1.1 Ementa da Disciplina

Curso: Curso Superior de Tecnologia em Sistemas para Internet **Disciplina:** Sistemas Embarcados **Carga-Horária:** 60h (80h/a) **Pré-Requisito(s):** Sistemas Digitais **Número de créditos:** 4

EMENTA

Aspectos relacionados ao desenvolvimento de sistemas embarcados.

PROGRAMA

Objetivos

Conhecer técnicas e ferramentas para desenvolvimento de Sistemas Embarcados.

Bases Científico-Tecnológicas (Conteúdos)

- 1. Introdução
 - 1.1. Histórico e evolução;
 - 1.2. Características;
 - 1.3. Aplicações típicas;
 - 1.4. Tecnologias e Arquiteturas;
 - 1.5. Projeto e Modelagem de Sistemas Embarcados.
- 2. Hardware
 - 2.1. Introdução aos microprocessadores e microcontroladores;
 - 2.2. Dispositivos de Entrada e Saída;
 - 2.3. Sensores;
 - 2.4. Atuadores;
 - 2.5. Interfaces de Comunicação.
- 3. Programação
 - 3.1. Ambientes de Desenvolvimento;
 - 3.2. Principais SOs para Sistemas Embarcados;
 - 3.3. Desenvolvimento de Sistemas Embarcados;
 - 3.4. Conectividades;

- 3.5. Programação concorrente: Conceitos de concorrência, problema de exclusão mútua, comunicação e sincronização em memória compartilhada e por troca de mensagens;
- 3.6. Escalonamento em projetos de sistemas embarcados;
- 3.7. Segurança.

Procedimentos Metodológicos

Aulas expositivas; aulas práticas, estudos dirigidos; seminários; vídeos; dinâmicas de grupo; visitas técnicas; palestras.

Recursos Didáticos

Quadro branco e pincel; computador; internet; projetor de multimídia.

Avaliação

Trabalhos práticos; apresentação de seminários; relatórios; avaliação escrita e prática.

Bibliografia Básica

- 1. ALMEIDA, Rodrigo de; MORAES, Carlos; SERAPHIM, Thatyana. Programação de Sistemas Embarcados. Editora Elsevier. 2016.
- 2. DENARDIN, Gustavo Weber; BARRIQUELLO, Carlos Henrique. Sistemas Operacionais de Tempo Real e sua Aplicação em Sistemas Embarcados. Editora Blucher. 2019.
- 3. JUNIOR, Sérgio Luiz Stevan; SILVA, Rodrigo Adamshuk. Automação e Instrumentação Industrial com Arduino: Teoria e Projetos. Editora Érica. 2015.

Bibliografia Complementar

- 1. YIU, Joseph. The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors. 3. ed. Newnes. 2013.
- 2. TOULSON, Rob; WILMSHURST, Tim. Fast and Effective Embedded Systems Design: Applying the ARM mbed. Editora Newnes. 2016.
- 3. JUCA, Sandro; PEREIRA, Renata. Aplicações Práticas de sistemas embarcados Linux utilizando Raspberry Pi. Editora PoD. 2018.
- 4. GU, Changyi. Building Embedded Systems: Programmable Hardware. Editora: Apress. 2016.
- 5. BERGER, Arnold S. Embedded Systems Design: An Introduction to Processes, Tools, and Techniques. Editora CMP Books. 2017.

Software(s) de Apoio:

- Linguagem de Programação;
- IDE - Ambiente de Desenvolvimento Integrado.

1.2 Introdução aos sistemas embarcados

Um sistema embarcado é:

“A combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a dedicated function. In some cases, embedded systems are part of a larger system or product, as in the case of an antilock braking system in a car.” – Michael Barr

Exemplos de sistemas embarcados são:

“Microwave ovens, cell phones, calculators, digital watches, VCRs, cruise missiles, GPS receivers, heart monitors, laser printers, radar guns, engine controllers, digital cameras, traffic lights, remote controls, bread machines, fax machines, pagers, cash registers, treadmills, gas pumps, credit/debit card readers, thermostats, pacemakers, blood gas monitors, grain analyzers, and a gazillion others.” – Michael Barr

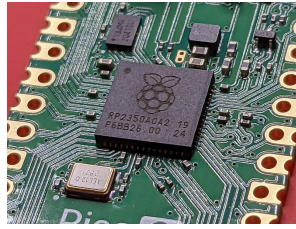


Figure 1: This is the RP2350 microcontroller chip seen in the circuit board of a Raspberry Pi Pico 2.

1.3 Introdução a microcontroladores

Um microcontrolador é:

“A highly integrated microprocessor designed specifically for use in embedded systems. Microcontrollers typically include an integrated CPU, memory (a small amount of RAM, ROM, or both), and other peripherals on the same chip. Common examples are Microchip’s PIC, the 8051, Intel’s 80196, and Motorola’s 68HCxx series.” – Michael Barr

1.4 Conceitos e fundamentos

...

2 Aula - 15/09/2025

Conteúdos ministrados: Conceitos básicos sobre sistemas embarcados - entradas digitais.

2.1 Entradas Digitais

...

3 Aula - 18/09/2025

Conteúdos ministrados: Introdução à programação de microcontroladores em linguagem C: tipos de dados; tabela ASCII e sistema binário; configuração de GPIO.

3.1 Introdução à programação de microcontroladores em linguagem C

...

3.1.1 Tipos de dados

...

3.1.2 Tabela ASCII e Sistema Binário

...

3.1.3 Configuração de GPIO

...

4 Aula - 22/09/2025

Conteúdos ministrados: Manipulando saídas digitais: Acionamento de LED; corrente máxima nos pinos; cálculo da resistência limitadora. Prática rápida: Implementação de um circuito de sinalização de entrada e saída de veículos. Interface entre circuitos digitais e cargas de alta potência utilizando transistores: Transistor como chave eletrônica; Acionamento de motores CC.

4.1 Manipulando saídas digitais

...

4.1.1 Acionamento de LED

...

4.1.2 Corrente máxima nos pinos

...

4.1.3 Cálculo da resistência limitadora

...

4.1.4 Prática rápida

...

Implementação de um circuito de sinalização de entrada e saída de veículos ...

Interface entre circuitos digitais e cargas de alta potência utilizando transistores ...

Transistor como chave eletrônica ...

Acionamento de motores CC ...

5 Aula - 25/09/2025

Conteúdos ministrados: Programação de microcontroladores: temporização. Funções delay() e millis(). Espera ocupada.

5.1 Programação de microcontroladores: Temporização

...

5.1.1 Função delay()

...

5.1.2 Função millis()

...

5.1.3 Espera ocupada

...

6 Aula - 29/09/2025

Conteúdos ministrados: Prática sobre acinamento de cargas utilizando transistores. Os alunos tiveram um primeiro contato com o laboratório de eletrônica, onde montaram um circuito simples de ativação de uma ventoinha por meio do chaveamento de um transistor de potência. O intuito da aula foi preparar a turma para futuras práticas de acionamento utilizando programação e transistores. Trabalhou-se a competência de se interpretar um esquema eletrônico, bem como de criar novos arranjos de acordo com uma necessidade específica.

6.1 Prática sobre acinamento de cargas utilizando transistores

...

7 Aula - 02/10/2025

Conteúdos ministrados: Programação com temporização: função millis().

7.1 Programação com temporização: função millis()

...

8 Aula - 06/10/2025

Conteúdos ministrados: Exercícios sobre entradas digitais, saídas, resistores de pull-up, pull-down, projeto de sistemas digitais utilizando arduino; Utilização dos operadores lógicos da linguagem C.

8.1 Exercícios

...

8.1.1 Entradas digitais

...

8.1.2 Saídas digitais

...

8.1.3 Resistores de pull-up

...

8.1.4 Resistores de pull-down

...

8.1.5 Projeto de sistemas digitais utilizando arduino

...

8.1.6 Utilização dos operadores lógicos da linguagem C

...

9 Aula - 09/10/2025

Conteúdos ministrados: Programação com temporização: função millis(). Aplicações.

9.1 Programação com temporização: função millis(). Aplicações

...

10 Aula - 13/10/2025

Conteúdos ministrados: Acionamento de cargas utilizando relé: Princípio de funcionamento dos relés eletromecânicos; simbologia; exemplo de aplicação.

10.1 Acionamento de cargas utilizando relé eletromecânicos

...

10.1.1 Princípio de funcionamento

...

10.1.2 Simbologia

...

10.1.3 Exemplo de aplicação

...

11 Atividade - 17/10/2025

Descrição da Atividade: Considerando o contexto da aula do dia 13/10, que tratou sobre os relés, faça o que se pede nos itens abaixo:

1. Implemente um código simples que inverte o estado da lâmpada caso um botão seja pressionado.

- Seguir esquema mostrado em aula;
- Materiais utilizados: Arduino Uno, pushbutton, relé SPDT, resistor, diodo, lâmpada incandescente, transistor NPN e bateria de 9V.

2. Envie o link da simulação.

11.1 Código em C++

```
// atribuição dos pinos
const int btn_pin = 2;
const int tst_pin = 3;

// atribuindo timestamp inicial
unsigned long millisNow = millis();
unsigned long lastMillis = millis();

// função para ler botão
void readButton(int time){
    millisNow = millis();
    if (millisNow - lastMillis >= time){
        digitalWrite(tst_pin, !digitalRead(tst_pin));
        lastMillis = millisNow;
    }
}
```

```

// configuração inicial
void setup()
{
    pinMode(btn_pin, INPUT);
    pinMode(tst_pin, OUTPUT);
    digitalWrite(btn_pin, LOW);
    digitalWrite(tst_pin, LOW);
}

// loop principal
void loop()
{
    if (digitalRead(btn_pin)){
        readButton(200); // duplo clique no intervalo de 200ms é considerado como um único clique
    }
}

```

12 Simulado da 1ª avaliação - 20/10/2025

...

13 Atividade: Genius da Estrela - 23/10/2025

...

References