



# Universidade Federal de Santa Catarina

Departamento de Computação

## IMPLEMENTAÇÃO DE UM ACELERADOR ELETRÔNICO PARA O PIC16F877A

Prof.: Rodrigo Pereira

Ítalo Silva  
Marcos Vinícius

Araranguá, 24 de junho de 2024

# Sumário

<b>1</b>	<b>Especificação de Produto</b>	<b>3</b>
<b>2</b>	<b>Requisitos técnicos</b>	<b>3</b>
<b>3</b>	<b>Descrição Resumida</b>	<b>3</b>
<b>4</b>	<b>Visão Geral</b>	<b>3</b>
<b>5</b>	<b>Funcionalidades</b>	<b>4</b>
5.1	Modo Normal . . . . .	4
5.2	Modo Turbo . . . . .	4
<b>6</b>	<b>Requisitos Funcionais</b>	<b>4</b>
<b>7</b>	<b>Requisitos não Funcionais</b>	<b>5</b>
<b>8</b>	<b>Requisitos de Testes</b>	<b>5</b>
<b>9</b>	<b>Projeto de Hardware</b>	<b>6</b>
9.1	Diagrama de blocos . . . . .	6
9.2	Tabela de interface . . . . .	7
<b>10</b>	<b>Projeto de Firmware</b>	<b>7</b>
10.1	Source Code . . . . .	7
10.2	Configurações . . . . .	11
<b>11</b>	<b>Fluxo de Funcionamento</b>	<b>12</b>
<b>12</b>	<b>Conclusão</b>	<b>13</b>

## 1 Especificação de Produto

- Projeto: Acelerador Eletrônico
- Responsáveis: Ítalo Silva e Marcos Vinícius
- Versão: 1.0
- Categoria: Gerenciador de performance para veículos automotores

## 2 Requisitos técnicos

- Microcontrolador: PIC16F877A
- Display LCD: 16x2 caracteres
- Potenciômetro: Sensor de posição do pedal do acelerador
- LED indicador: Para sinalizar os modos Normal e Turbo

## 3 Descrição Resumida

O sistema de Controle de Modo Normal e Turbo de um Acelerador Eletrônico foi desenvolvido para implementar controlador capaz de alternar entre dois modos de operação: Normal e Turbo. Este sistema tem como objetivo proporcionar maior controle sobre a potência do veículo, permitindo ao condutor selecionar entre um modo de condução mais econômico e um modo de alta performance, ideal para situações de ultrapassagem.

## 4 Visão Geral

O Acelerador Eletrônico é um sistema projetado para veículos, permitindo alternar entre modos de operação Normal e Turbo, oferecendo flexibilidade e controle ao condutor. A aplicação alvo deste projeto são veículos automotivos, onde há a necessidade de controlar a potência do motor de acordo com a situação de condução. Este sistema usa sensores para interagir com o acelerador do veículo e possui interface própria de feedback visual.

## 5 Funcionalidades

### 5.1 Modo Normal

- Neste modo, o acelerador alcança até 100% de sua capacidade máxima.
- Indicado para condução cotidiana e econômica.
- O LED indicador de modo Normal permanece aceso.

### 5.2 Modo Turbo

- Neste modo, o acelerador pode alcançar até 115% de sua capacidade máxima.
- Ideal para situações que demandam uma resposta mais rápida do veículo, como ultrapassagens.
- Após acionado, o modo Turbo é temporário, sendo automaticamente desativado após 20 segundos para evitar estresse térmico no sistema.
- Durante os primeiros 15 segundos de operação do modo Turbo, o LED indicador de modo Turbo permanece aceso.
- Nos últimos 5 segundos de operação do modo Turbo, o LED indicador pisca, indicando que o sistema voltará ao modo Normal.

## 6 Requisitos Funcionais

- RF01: O sistema deve permitir a alternância entre o modo de operação Normal e Turbo.
- RF02: O sistema deve ajustar a resposta do acelerador de acordo com o modo selecionado, proporcionando uma condução econômica no modo Normal e alta performance no modo Turbo.
- RF03: O sistema deve fornecer feedback visual ao condutor sobre o modo de operação atual.
- RF04: O sistema deve incluir uma função de segurança que desative o modo Turbo após o tempo limite de utilização.

## 7 Requisitos não Funcionais

- RNF01: O sistema terá um botão de seleção que permitirá ao condutor alternar entre o modo Normal e o modo Turbo.
- RNF02: O sistema terá um potenciômetro que mapeará a posição do pedal.
- RNF03: Para informar ao condutor sobre o modo de operação atual, o sistema utilizará indicadores visuais de modo de operação (LED) e posição do acelerador (LCD).
- RNF04: Para evitar estresse térmico o sistema realizará automaticamente a transição de volta para o modo Normal. Esse controle automático será implementado através de um temporizador no software (Timer 1) do sistema, que é acionado sempre que o modo Turbo é ativado.

## 8 Requisitos de Testes

- RT01: Testar a alternância entre os modos de operação Normal e Turbo.

Descrição: Acionar o sistema para alternar repetidamente entre os modos Normal e Turbo.

Critério de Aceitação: Verificar se o sistema responde corretamente à seleção do modo.

- RT02: Verificar o ajuste de amplitude do acelerador de acordo com o modo selecionado:

Descrição: Ajustar manualmente o modo entre Normal e Turbo e observar o comportamento do acelerador.

Critério de Aceitação: Verificar se a resposta do acelerador está de acordo com o modo selecionado.

- RT03: Validar o feedback visual ao condutor sobre o modo de operação atual:

Descrição: Observar os indicadores visuais fornecidos pelo sistema durante a alternância entre os modos.

Critério de Aceitação: Verificar se o condutor recebe feedback claro e imediato sobre o modo de operação atual do acelerador.

- RT04: Testar a transição automática para o modo Normal após 20 segundos de operação no modo Turbo para evitar superaquecimento:

Descrição: Ativar o modo Turbo e cronometrar o tempo de operação.

Critério de Aceitação: Verificar se o sistema realiza a transição automática para o modo Normal após 20 segundos de operação contínua no modo Turbo, demonstrando a funcionalidade de proteção contra superaquecimento.

- Link para os testes: <https://www.youtube.com/watch?v=X-Qi25bYLIE>

## 9 Projeto de Hardware

### 9.1 Diagrama de blocos

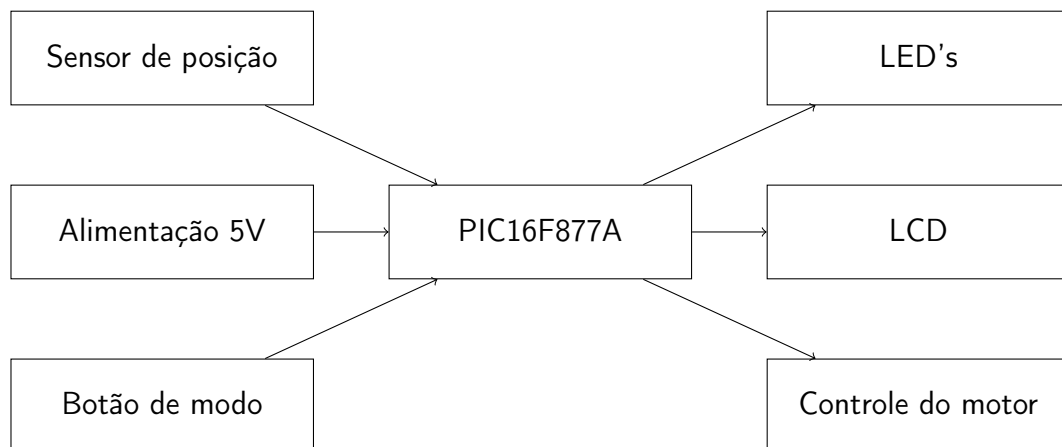


Diagrama de Blocos do Sistema de Acelerador Eletrônico

## 9.2 Tabela de interface

PIC16F877A	Entrada/Saída	Nível lógico	Descrição
LED	Saída	1	Diodo emissor de luz
Regulador de tensão	Saída	5V	Ajuste de tensão
Display LCD	Saída	4,5V/5,5V	Tela de cristal líquido
Potenciômetro	Entrada	0V/12V	Resistor de resistência variável
Botão de modo	Entrada	1	Botão
Botão de reset	Entrada	1	Botão
Cristal de Quartzo	Entrada	0/1	Oscilador
LED	Saída	<0,8V/>2,2V	Entrada de nível do LED
Regulador de tensão	Entrada	12V	Barramento do veículo
	Saída	5V	Alimentação do PIC16F877A
Display LCD	Entrada	1	Dados do PIC16F877A
	Saída	1	Tela de cristal líquido
Potenciômetro	Entrada	0V/12V	Sensor de posição do pedal
Botão de modo	Entrada	0/1	Seleção de modo
Botão de reset	Entrada	0/1	Reset do sistema
Cristal de Quartzo	Entrada	0/1	Frequência do ciclo de instrução

## 10 Projeto de Firmware

### 10.1 Source Code

O firmware desenvolvido pode ser analisado em sua íntegra abaixo

```

1 #pragma config FOSC = HS
2 #pragma config WDTE = OFF
3 #pragma config PWRTE = OFF
4 #pragma config BOREN = ON
5 #pragma config LVP = OFF
6 #pragma config CPD = OFF
7 #pragma config WRT = OFF
8 #pragma config CP = OFF
9
10 #include <stdio.h>
11 #include <stdbool.h>
12 #include <xc.h>
13 #include "lcd.h"
14
15 #define _XTAL_FREQ 4000000
16 #define NORMAL_PIN RC3
17 #define TURBO_PIN RC4;

```

```

18
19 volatile unsigned char mode = 0;
20 volatile unsigned long int turbo_timer = 0;
21 volatile bool turbo_active = false;
22 volatile unsigned int timer_counter = 0;
23 volatile float acel_percent;
24
25 void init_ADC() {
26     ADCON0 = 0x41;
27     ADCON1 = 0x80;
28 }
29
30 unsigned int read_ADC() {
31     ADCON0bits.GO_DONE = 1;
32     while (ADCON0bits.GO_DONE);
33     return ((unsigned int)(ADRESH << 8) + ADRESL);
34 }
35
36 void init_interrupt() {
37     INTCONbits.GIE = 1;
38     INTCONbits.PEIE = 1;
39     INTCONbits.INTE = 1;
40     OPTION_REGbits.INTEDG = 0;
41     PIE1bits.TMR1IE = 1;
42 }
43
44 void init_timer(){
45     T1CONbits.TMR1CS = 0;
46     T1CONbits.T1CKPS0 = 1;
47     T1CONbits.T1CKPS1 = 1;
48     TMR1L = 0xDC;
49     TMR1H = 0x0B;
50     T1CONbits.TMR1ON = 1;
51 }
52
53 void __interrupt() ISR() {
54     if (INTCONbits.INTF) {
55         mode = !mode;
56         INTCONbits.INTF = 0;
57     }
58
59     if (PIR1bits.TMR1IF) {
60         PIR1bits.TMR1IF = 0;
61         TMR1L = 0xDC;
62         TMR1H = 0x0B;
63
64         if (turbo_active) {
65             turbo_timer++;
66             if (turbo_timer >= 20) {

```



```

67         turbo_active = false;
68         turbo_timer = 0;
69         timer_counter = 10;
70     }
71     } else if (timer_counter > 0) {
72         timer_counter--;
73         TURBO_PIN = !TURBO_PIN;
74         __delay_ms(500);
75         if (timer_counter == 0) {
76             mode = 0;
77             TURBO_PIN = 0;
78         }
79     }
80 }
81 }
82
83 int main(void) {
84     TRISC = 0x00;
85     NORMAL_PIN = 1;
86
87     LCD lcd = { &PORTD, 2, 3, 4, 5, 6, 7 };
88     LCD_Init(lcd);
89     init_interrupt();
90     init_ADC();
91     init_timer();
92
93     char buffer[16];
94
95     while(1) {
96         unsigned int adc_value = read_ADC();
97
98         if (mode == 0) {
99             NORMAL_PIN = 1;
100             TURBO_PIN = 0;
101             acel_percent = (adc_value / 889.56f) * 100;
102             if (acel_percent > 100) acel_percent = 100;
103         } else {
104             TURBO_PIN = 1;
105             NORMAL_PIN = 0;
106             acel_percent = (adc_value / 1023.0f) * 115;
107             if (acel_percent > 115) acel_percent = 115;
108             if (acel_percent == 115 && !turbo_active &&
timer_counter == 0) {
109                 turbo_active = true;
110                 turbo_timer = 0;
111             }
112         }
113
114         LCD_Clear();

```

```

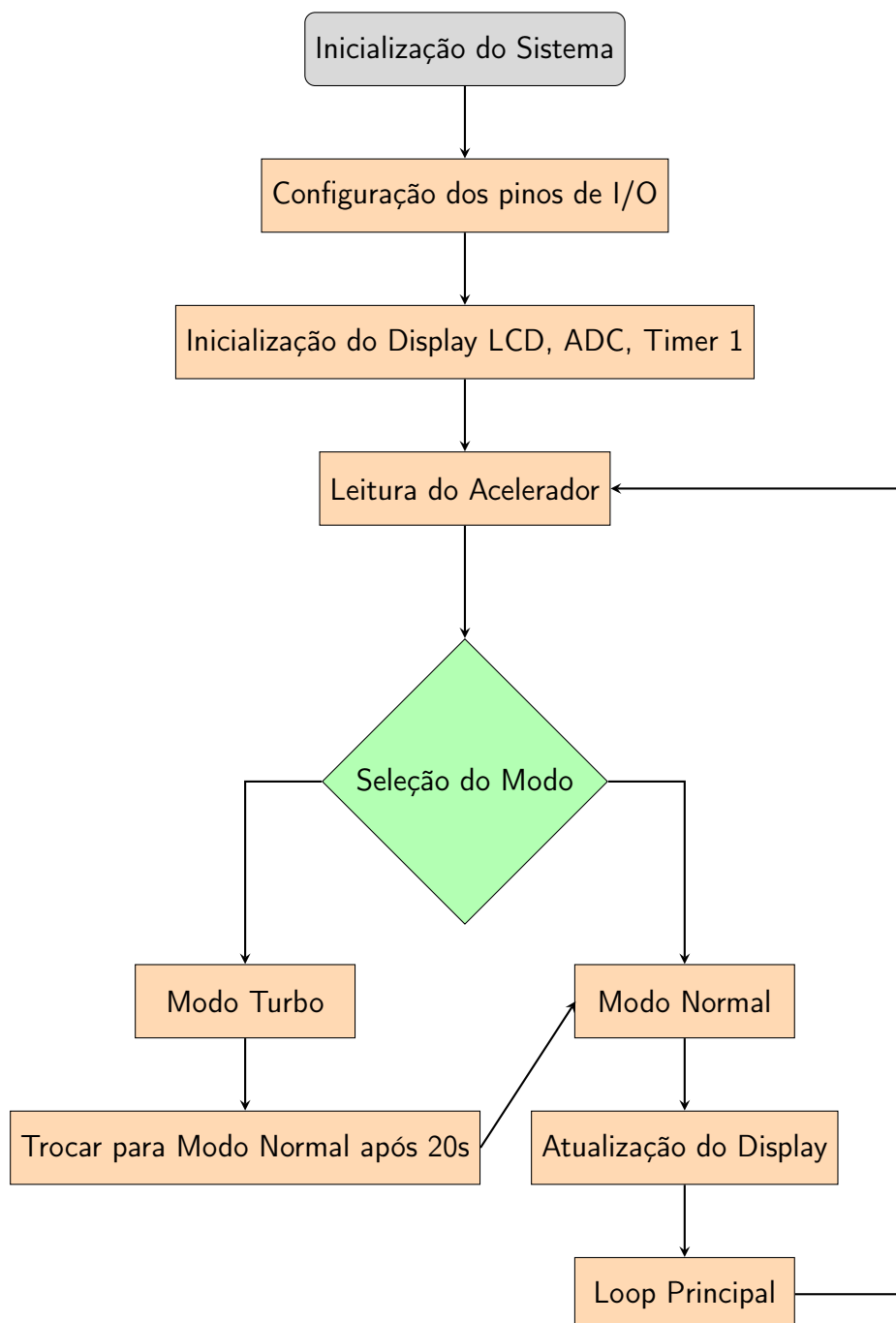
115         LCD_Set_Cursor(0, 0);
116         sprintf(buffer, "Acel: %.1f%%", acel_percent);
117         LCD_puts(buffer);
118
119
120         LCD_Set_Cursor(1, 0);
121         if (mode == 0) {
122             LCD_puts("Modo: Normal");
123         } else {
124             LCD_puts("Modo: Turbo");
125         }
126
127         __delay_ms(200);
128     }
129
130     return (EXIT_SUCCESS);

```

## 10.2 Configurações

- Oscilador: HS (High-Speed)
- Watchdog Timer: Desativado durante o desenvolvimento e testes, pode ser ativado para garantir maior segurança em situações de operação crítica.
- Power-up Timer: Desativado
- Brown-out Reset: Ativado para garantir maior robustez contra falhas de alimentação.
- Low-Voltage Programming: Desativado
- Proteção de Memória: Desativada tanto para a EEPROM quanto para a Flash Program Memory.

## 11 Fluxo de Funcionamento



## 12 Conclusão

O sistema de Controle de Modo Normal e Turbo de um Acelerador Eletrônico proporciona uma maneira eficiente e segura de controlar a potência de um veículo, oferecendo ao condutor a flexibilidade de escolher entre diferentes modos de condução conforme a necessidade. Com sua implementação robusta e recursos avançados, este sistema contribui para uma experiência de direção mais confortável e personalizada.