Anexo A4 – Controle de Tempo com Botão (led\_dinamico\_btn)

# Descrição

Este exemplo amplia o modelo cooperativo com callback, permitindo alterar o tempo de piscar do LED usando um botão. A frequência de alternância é controlada dinamicamente e o tempo de visibilidade do LED é proporcional ao novo período.

# Objetivo

- Utilizar add\_alarm\_in\_ms() com callback periódico  
- Alternar entre períodos de 1000 ms e 500 ms ao pressionar um botão  
- Tornar o piscar do LED visível ao ajustar o tempo de acionamento

# Código-fonte Corrigido (main.c)

#include <stdio.h>  
#include "pico/stdlib.h"  
#include "hardware/timer.h"  
  
#define LED\_PIN 12  
#define BUTTON\_PIN 5  
  
volatile bool alternar\_led = false;  
bool estado\_led = false;  
uint32\_t periodo\_ms = 1000;  
bool botao\_anterior = true;  
  
int64\_t temporizador\_callback(alarm\_id\_t id, void \*user\_data) {  
 alternar\_led = true;  
 return periodo\_ms;  
}  
  
int main() {  
 stdio\_init\_all();  
  
 gpio\_init(LED\_PIN);  
 gpio\_set\_dir(LED\_PIN, GPIO\_OUT);  
  
 gpio\_init(BUTTON\_PIN);  
 gpio\_set\_dir(BUTTON\_PIN, GPIO\_IN);  
 gpio\_pull\_up(BUTTON\_PIN);  
  
 add\_alarm\_in\_ms(periodo\_ms, temporizador\_callback, NULL, true);  
  
 while (true) {  
 bool botao\_atual = gpio\_get(BUTTON\_PIN);  
  
 if (botao\_anterior && !botao\_atual) {  
 periodo\_ms = (periodo\_ms == 1000) ? 500 : 1000;  
 printf("Novo período: %u ms\n", periodo\_ms);  
 }  
 botao\_anterior = botao\_atual;  
  
 if (alternar\_led) {  
 estado\_led = !estado\_led;  
 gpio\_put(LED\_PIN, estado\_led);  
 printf("LED = %d\n", estado\_led);  
 sleep\_ms(periodo\_ms / 4);  
 alternar\_led = false;  
 }  
  
 tight\_loop\_contents();  
 }  
}

# CMakeLists.txt (Projeto: led\_dinamico\_btn)

cmake\_minimum\_required(VERSION 3.13)  
include(pico\_sdk\_import.cmake)  
  
project(led\_dinamico\_btn C CXX ASM)  
pico\_sdk\_init()  
  
add\_executable(led\_dinamico\_btn  
 main.c  
)  
  
target\_link\_libraries(led\_dinamico\_btn pico\_stdlib hardware\_timer)  
pico\_enable\_stdio\_usb(led\_dinamico\_btn 1)  
pico\_enable\_stdio\_uart(led\_dinamico\_btn 0)  
  
pico\_add\_extra\_outputs(led\_dinamico\_btn)

# Observações Práticas

- O botão altera entre dois períodos de temporização (500 ms e 1000 ms).  
- O tempo de visibilidade do LED é proporcional ao novo período.  
- A borda de descida é usada para evitar múltiplas trocas por ruído.  
- O terminal exibe a frequência e o estado do LED em tempo real.