Anexo A5 – Multitarefa com Dois Núcleos (multicore\_led\_mensagem)

# Descrição

Este exemplo demonstra a utilização dos dois núcleos do RP2040 para executar tarefas simultâneas. O Core 0 é responsável por controlar o piscar de um LED, enquanto o Core 1 imprime mensagens no terminal a cada segundo.

# Objetivo

- Executar tarefas simultâneas com multicore  
- Piscar LED (GPIO 12) no Core 0  
- Imprimir mensagens no terminal via Core 1

# Código-fonte (main.c)

#include <stdio.h>  
#include "pico/stdlib.h"  
#include "pico/multicore.h"  
  
#define LED\_PIN 12  
  
void tarefa\_core1() {  
 while (true) {  
 printf("Tarefa no Core 1 ativa\n");  
 sleep\_ms(1000);  
 }  
}  
  
int main() {  
 stdio\_init\_all();  
  
 gpio\_init(LED\_PIN);  
 gpio\_set\_dir(LED\_PIN, GPIO\_OUT);  
  
 multicore\_launch\_core1(tarefa\_core1);  
  
 while (true) {  
 gpio\_put(LED\_PIN, 1);  
 sleep\_ms(500);  
 gpio\_put(LED\_PIN, 0);  
 sleep\_ms(500);  
 }  
}

# CMakeLists.txt (Projeto: multicore\_led\_mensagem)

... (conteúdo baseado na estrutura enviada anteriormente com pico\_multicore incluído) ...

# Observações Práticas

- O LED pisca em uma frequência constante (500ms ligado, 500ms desligado).  
- O segundo núcleo (Core 1) imprime mensagens no terminal paralelamente.  
- Os núcleos não interferem entre si, demonstrando multitarefa real por hardware.  
- Este exemplo pode ser estendido com comunicação entre núcleos usando FIFO (ver Anexo A6).