**Ficheiros de projeto:**

GMTpwm.c: Este ficheiro contém a implementação de funções relacionadas à saída PWM (Modulação por Largura de Pulso), fornecendo a funcionalidade necessária para escrever sinais PWM para o pino de saída designado.

GMTadc.c: Este ficheiro inclui a implementação de funções relacionadas com as entradas ADC (Analog-to-Digital Conversion). Permite ler o estado das entradas analógicas e atualizar os seus valores.

rtdb.c: Este ficheiro implementa a base de dados em tempo real (RTDB), que serve de memória partilhada para armazenar e recuperar dados entre diferentes threads. Fornece funções para aceder e atualizar dados na base de dados.

main.c: Este é o ficheiro principal do projeto. Orquestra a execução de diferentes threads, configura os pinos de entrada/saída, ADC, PWM e trata a entrada de comandos via UART. Configura a RTDB, cria e gere as threads e contém a função principal para executar o programa.

**Tópicos:**

Thread Imprimir:

Prioridade: thread\_print\_prio (predefinição: 3)

Período: thread\_print\_period (predefinição: 1000 ms)

Descrição: Esta thread lê periodicamente o estado das entradas analógicas e actualiza os seus estados na RTDB.

Thread de Entrada Analógica:

Prioridade: thread\_an\_prio (predefinição: 3)

Período: thread\_an\_period (predefinição: 1000 ms, pode variar via UART)

Descrição: Esta thread lê periodicamente o estado das entradas analógicas e actualiza os seus valores na RTDB.

Thread PWM:

Prioridade: thread\_pwm\_prio (predefinição: 3)

Período: thread\_pwm\_period (predefinição: 1000 ms)

Descrição: Esta thread escreve periodicamente sinais PWM no pino de saída designado com base na informação presente na RTDB.

Thread de comandos:

Prioridade: thread\_cmd\_prio (predefinição: 4)

Período: thread\_cmd\_period (predefinição: 1000 ms)

Descrição: Esta thread é activada sempre que é introduzido um comando via UART. Procura comandos válidos no buffer de comandos e executa as acções correspondentes, tais como alterar os períodos de outras threads.

**Prioridades e Escalonamento:**

A prioridade de cada thread determina a ordem em que elas são agendadas para execução. Um valor de prioridade mais alto indica um nível de prioridade mais alto. Na configuração padrão, todos os threads têm a mesma prioridade, exceto o Thread de comando, que tem uma prioridade um pouco mais alta.

Com o período e as prioridades padrão, o agendamento das threads ocorreria da seguinte forma:

Thread Comandos: Seria agendada primeiro devido à sua prioridade mais elevada, permitindo-lhe processar prontamente quaisquer comandos de entrada.

Thread de entrada analógica, Thread de impressão e Thread PWM: Estas threads seriam programadas de forma round-robin, com igual prioridade e períodos. Cada thread obteria uma parte justa do tempo de CPU para execução.

Se o utilizador alterar os períodos das threads, elas entrarão na ready-queue em tempos diferentes, logo, deixa de funcionar por round-robin.

**Acesso a recursos partilhados:**

Foi previsto o acesso a recursos partilhados, controlado via semáforos.

**Comandos válidos:**

Para alterar os períodos das threads, deve ser introduzido um comando via UART no seguinte formato:

Formato do comando: $TXYYYY& (ou $tXYYYY&)

X: Pode ser 'O' ou 'o' para se referir à thread PWM, ou 'I' ou 'i' para se referir à thread de entrada analógica.

Y: Representa quatro dígitos que especificam o tempo em milissegundos (ms).

Exemplo: $O2000& define o período da Thread PWM para 2000 ms.

A entrada de comando tem de seguir o formato supramencionado para ser considerada válida.