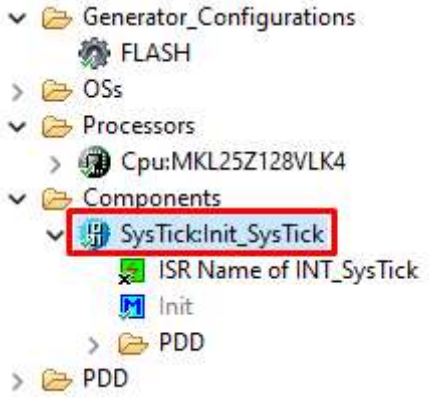
PSI3441 - Arquitetura de Sistemas Embarcados

Relatório Exercício Prático 03

Nome: Kevin Kirsten Lucas nºUSP: 10853306

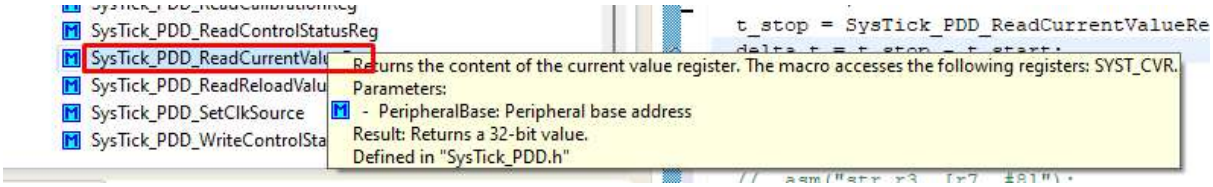
1) Compile e rode o código abaixo no CodeWarrior e use o SysTick para medir o tempo de execução deste código: $c = a + b$ *sendo* $a = 2$ *e* $b = 2$

Para essa primeira tarefa, é necessário utilizar o SysTick, para isso, precisamos adicionar o componente Init_SysTick, como mostrado abaixo.



Componente Init_SysTick

Para ler os valores do registrador com a contagem SysTick, basta utilizar o atributo `SysTick_PDD_ReadCurrentValueReg`.



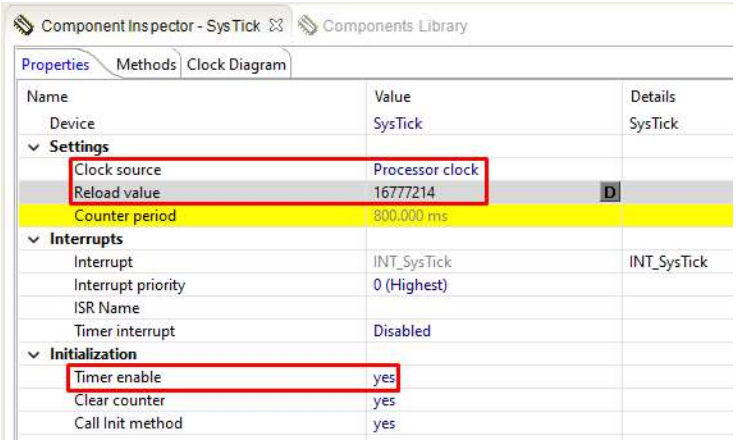
Leitura do valor do registrador SysTick

Acho válido ressaltar que pode ser necessário também alterar o include que o Processor Expert faz. Perdi alguns bons dias por não saber que era necessário adicionar o import:

```
#include "SysTick_PDD.h"
```

No início do `main.cpp`.

Por último, basta configurar um valor arbitrário para o Reload value, e habilitar o timer:



Reload value

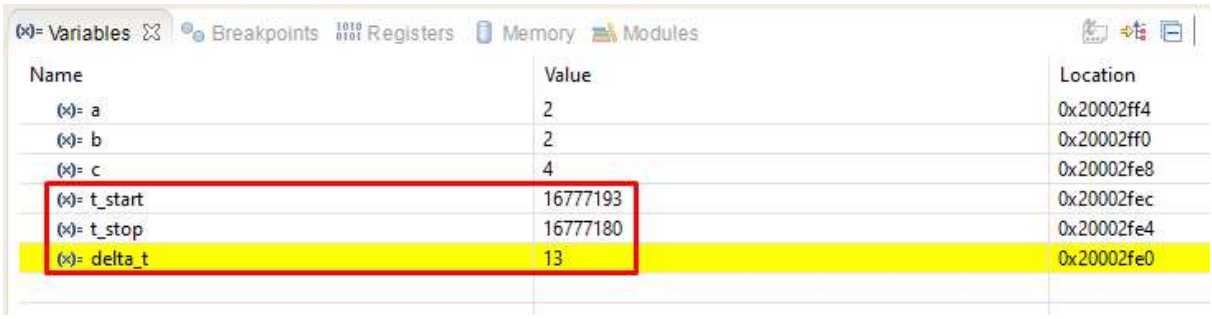
Com isso, o código final, ficou como mostrado abaixo.

Obs, a linha 10 foi utilizada como breakpoint para observar os valores em cada uma das variáveis.

```
1  int t_start, t_stop, delta_t;
2  int a, b, c;
3  a = 2;
4  b = 2;
5
6  t_start = SysTick_PDD_ReadCurrentValueReg(SysTick_DEVICE);
7  c = a + b;
8  t_stop = SysTick_PDD_ReadCurrentValueReg(SysTick_DEVICE);
9
10 delta_t = t_start - t_stop; // breakpoint
```

Código da Tarefa 1

Após executada a simulação, temos o seguinte resultado nos registradores:



Valores dos registradores após execução do processador

O tempo de execução pode ser calculado da seguinte forma:

`delta_t de 1 tick = 800 ms / 16777214 ticks`

`delta_t de 1 tick * 13 ticks = 0.62 µs`

2) Analise o código em Assembly (Disassembly) gerado pelo CodeWarrior e descreva cada uma das instruções. Utilize o simulador VisUAL para auxilia-lo.

Abaixo podemos ver o código de Disassembly gerado pelo CodeWarrior

```
62          t_start = SysTick_PDD_ReadCurrentValueReg(SysTick_DEVICE);
0000077e:   ldr r3,[pc,#28]
00000780:   ldr r3,[r3,#8]
00000782:   str r3,[r7,#12]
63          c = a + b;
00000784:   ldr r2,[r7,#20]
00000786:   ldr r3,[r7,#16]
00000788:   adds r3,r2,r3
0000078a:   str r3,[r7,#8]
66          t_stop = SysTick_PDD_ReadCurrentValueReg(SysTick_DEVICE);
0000078c:   ldr r3,[pc,#12]
0000078e:   ldr r3,[r3,#8]
00000790:   str r3,[r7,#4]
68          delta_t = t_start - t_stop;
00000792:   ldr r2,[r7,#12]
00000794:   ldr r3,[r7,#4]
00000796:   subs r3,r2,r3
00000798:   str r3,[r7,#0]
```

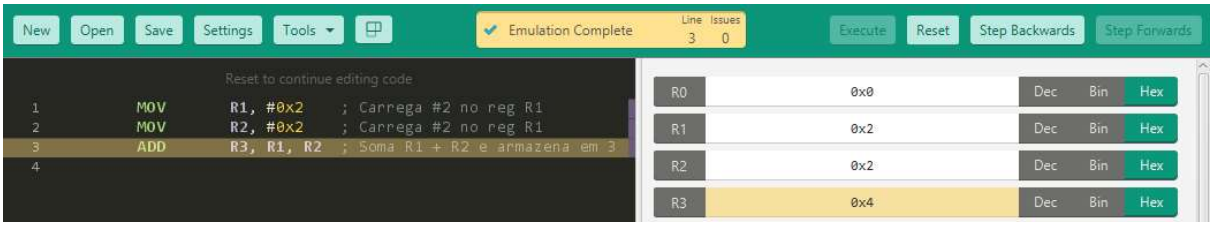
Código Disassembly

A seguir, temos a descrição das partes do código.

```
@ t_start = SysTick_PDD_ReadCurrentValueReg(SysTick_DEVICE);
ldr r3,[pc, #28] @ r3 carregado com o endereço de program counter + offset de 28
ldr r3,[r3,  #8] @ r3 carregado com o valor de r3 + offset de 8
str r3,[r7, #12] @ r3 armazenado no endereço de r7 + offset de 12
@ c = a + b;
ldr  r2,[r7,#20] @ r2 carregado com o valor de r7 + offset de 20
ldr  r3,[r7,#16] @ r3 carregado com o valor de r7 + offset de 16
adds r3, r2, r3  @ r3 = r2 + r3 e armazena o resultado em r3
str  r3,[r7, #8] @ r3 armazenado no endereço de r7 + offset de 8
@ t_stop = SysTick_PDD_ReadCurrentValueReg(SysTick_DEVICE);
ldr  r3,[pc,#12] @ r3 carregado com o endereço de program counter + offset de 12
ldr  r3,[r3, #8] @ r3 carregado com o valor de r3 + offset de 8
str  r3,[r7, #4] @ r3 armazenado no endereço de r7 + offset de 4
@ delta_t = t_start - t_stop;
ldr  r2,[r7,#12] @ r2 carregado com o valor de r7 + offset de 12
ldr  r3,[r7, #4] @ r3 carregado com o valor de r7 + offset de 4
subs r3, r2, r3  @ r3 = r2 - r3 e armazena o resultado em r3
str  r3,[r7, #0] @ r3 armazenado no endereço de r7 + offset de 0
```

Descrição do Código Disassembly

3) Otimize o código no VisUAL para reduzir o número de ciclos necessários para executar a mesma operação.



Otimização do código no VisUAL

4) Implemente o código otimizado no CodeWarrior e meça o tempo para executa-lo. Utilize a sintaxe abaixo para escrever em assembly no CodeWarrior: asm("Operação Destino, Operando 1, Operando 2\n"); exemplo: asm("ADD R0, R1,R2\n") ; R0 = R1+R2

Podemos implementar o código do exercício 3, da seguinte forma no CodeWarrior

```
1   int t_start, t_stop, delta_t;
2
3   asm("mov r3, #2");
4   asm("mov r2, #2");
5   t_start = SysTick_PDD_ReadCurrentValueReg(SysTick_DEVICE);
6   asm("add r3, r2, r3");
7   asm("str r3, [r7, #8]");
8   t_stop = SysTick_PDD_ReadCurrentValueReg(SysTick_DEVICE);
9
10  delta_t = t_start - t_stop;
```

Implementação do código no CodeWarrior