

**INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA**  
**LEETC, LEIC, MEET, MEIC**  
**Sistemas Embebidos I**  
**Semestre de Inverno de 2010/2011**

## Terceira actividade prática

### Programação em C

Na geração de programas em C intervêm normalmente vários módulos escritos em C com o código específico do programa, o módulo de preparação para a execução, normalmente escrito em Assembly - **cstart.s**, o ldscript com a indicação de localização do programa final e um ficheiro makefile com as instruções de geração.

Os ficheiros **cstart.s**, **ldscript** e **makefile** constituem uma base comum para a geração de qualquer programa em C. O objectivo deste trabalho é a criação dessa base.

Colocar o seguinte programa em **main.c**

```
#include "led.h"

int main() {
    led_init();
    while (1) {
        int i;
        led_write(0);
        for (i = 0; i < 10000; ++i
            ;
        led_write(1);
        for (i = 0; i < 10000; ++i
            ;
    }
}
```

Programar a função **void led\_write(int)** colocando o código em **led.c** e a declaração em **led.h**.

Criar os ficheiros **cstart.s**, **ldscript** e **makefile**.

Consultar os apontamentos "Programação ARM com ferramentas GNU".

```
sel
|
+-- code
|   +-- app
|       +-- led
|           +-- main.c
|           +-- makefile
|       +-- mylib
|           +-- cstart.s
|           +-- ldscript
|           +-- led.c
|           +-- led.h
```

## Relógio do sistema

O LPC2106 possui dois sinais de relógios principais o **CCLK** e o **PCLK** cuja frequência é programável em função de um oscilador a cristal. Esta programação é feita nos periféricos **Phase Locked Loop** e **APB Divider**.

Programar a função `void lpc2106_clock_init()` que, manipulando estes periféricos, define a frequência dos sinais CCLK e PCLK de acordo com as seguintes definições:

```
#define FOSC      14745600
#define PLL_M     4
#define PLL_P     2
#define CCLK      (FOSC * PLL_M)
#define PCLK      (CCLK / 4)
```

Criar o ficheiro `mylib/lpc2106.h` para conter as definições anteriores e outras relacionadas com o microcontrolador como endereços e máscaras de manipulação de periféricos.

Criar também o ficheiro `mylib/lpc2106.c` onde será colocado o código da função `lpc2106_clock_init` e outro código relacionado com o controlo do microcontrolador.

Como testar a função `lpc2106_clock_init`? Os sinais CCLK e PCLK não são acessíveis directamente. A forma que temos de os testar é por via indirecta. Para isso, usar-se-há o periférico *timer* tratado no ponto seguinte.

## Timer

Programar a função `void timer_init()` para colocar o registo contador do *timer* 0 a incrementar a uma frequência aproximada de 1 KHz.

Os periféricos *timer* usam como base de tempo o sinal PCLK definido no ponto anterior.

Programar a função `U32 timer_read()` para ler o valor corrente do contador do *timer* 0.

Colocar as funções anteriores em `lpc2106.c` e as respectivas declarações em `lpc2106.h`.

## Medir intervalos de tempo

A função `chrono_elapsed` cujo código se apresenta devolve o tempo decorrido desde um momento anterior, representado pelo parâmetro `initial`, e o momento actual.

```
U32 chrono_elapsed(U32 initial) {
    return timer_read() - initial;
}
```

Programar a função `void chrono_delay(U32 msec)` que, utilizando funções anteriores, aguarda a passagem do intervalo de tempo indicado em `msec` (a unidade é o milisegundo).

Usar o seguinte programa para teste:

```
#include "lpc2106.h"
```

```
#include "chrono.h"
#include "led.h"

int main() {
    lpc2106_init();
    chrono_init();
    led_init();
    while (1) {
        led_write(1);
        chrono_delay(500);
        led_write(0);
        chrono_delay(500);
    }
}
```

Com a ajuda de um relógio, este programa irá permitir verificar, embora grosseiramente, o funcionamento das funções desenvolvidas até agora.

Depois de concluída a actividade prática deve entregar ao docente a listagem dos programas realizados. Entregue por email e segundo a estrutura aconselhada em "Ambiente de trabalho".

29/9/2010  
Ezequiel Conde, Pedro Sampaio