

Ensino de Microcontroladores usando a família MSP430

Humberto Santos*, Pedro Dinis Gaspar*, António Espírito Santo* e Bruno Ribeiro*

* Departamento de Engenharia Electromecânica

Universidade da Beira Interior

Calçada Fonte do Lameiro, 6201-001 Covilhã

Telf: +351 275 329 925; fax: +351 275 329 972;

e-mail: humberto@ubi.pt, dinis@ubi.pt, aes@ubi.pt, bruno@ubi.pt

Resumo — Este artigo descreve o uso da família de microcontroladores MSP430 nas aulas dos cursos de Eng^a Electromecânica e Eng^a Electrotécnica. Esta família de microcontroladores foi escolhida devido às suas características e capacidades. O custo do μ C e dos sistemas de desenvolvimento, a disponibilidade de várias ferramentas de desenvolvimento e o grande número de periféricos que esta família de microcontroladores possui permite o desenvolvimento de aplicações com o enfoque desejado para as aulas. As capacidades de análise de erros do *software* e as ferramentas de programação em linguagem C permitem uma aprendizagem mais acessível por parte dos alunos.

Tendo em atenção as principais capacidades que queremos fornecer aos nossos estudantes, neste artigo vamos mostrar algumas das aplicações desenvolvidas e algumas em desenvolvimento pelo grupo de trabalho em microprocessadores do Departamento de Electromecânica da Universidade da Beira Interior.

O objectivo deste artigo é o de mostrar uma forma acessível para o ensino de microprocessadores fazendo uso de ferramentas simples, mas ao mesmo tempo, poderosas e completas, que podem ser usadas numa grande variedade de aplicações.

1. Introdução

Cabe às instituições de ensino superior colaborar no desenvolvimento tecnológico. Esta tarefa pode ser efectuada quer transferindo tecnologia para as indústrias que com elas colaboram, quer formando recursos humanos. Diversas disciplinas do plano curricular dos cursos de Engenharias leccionadas pelo Departamento de Engenharia Electromecânica (DEM) da Universidade da Beira Interior, têm vindo a implementar o ensino de sistemas microprocessados, baseados na família MSP430, por forma a dotar os alunos de conhecimentos profundos sobre este tipo de tecnologia. É necessário reconhecer que diversas multinacionais envolvidas no desenvolvimento e utilização deste tipo de tecnologias têm-se implantado em Portugal, sendo então aconselhável a formação de meios técnicos nestas áreas.

A utilização da família MSP430 deve-se fundamentalmente aos seguintes aspectos:

- Custo;
- Diversidade de ferramentas de desenvolvimento grátis;

- *Kits* de *hardware* de iniciação apropriados e com custo reduzido;
- Diversos periféricos no microprocessador que permitem dotarem os alunos de capacidades adicionais;
- Curva de aprendizagem rápida;
- Características de baixo consumo energético;
- Elevado número de pinagem I/O;
- Programação em ANSI C e em C++ ...

A escolha do *hardware* recaiu sobre os *kits* fornecidos pela OLIMEX [1], pelas razões já apontadas.

No mercado encontram-se vários compiladores destinados à configuração e programação dos *kits*. Entre estes, encontram-se os pacotes fornecidos pela TEXAS INSTRUMENTS (Code Composer [2]); IAR SYSTEMS (IAR Embedded Workbench - Kickstart Version [3]); e também o MSPGCC [4] que consiste num compilador desenvolvido pela *open-source community*.

Até aqui, têm sido utilizados os compiladores IAR SYSTEMS (IAR Embedded Workbench - Kickstart Version) e MSPGCC. Embora, sejam versões grátis e que possuem um ambiente de desenvolvimento integrado para a compilação, *assembler*, *debugger* e simulação de aplicações embebidas em linguagem *assembler*, C e C++, o tamanho máximo de código que possuem é 4 kBytes. Assim, neste momento está-se a transitar para o *Code Composer* por permitir o desenvolvimento de códigos até 8 kBytes.

2. MSP@UBI

Esta família de microprocessadores é utilizada em várias disciplinas dos cursos de Engenharia Electromecânica e Engenharia Electrotécnica, requerendo no entanto vários conceitos base por parte dos alunos e já interiorizados em disciplinas precedentes, tais como: conhecimentos de electrónica analógica/digital; programação em *assembler*, C e/ou C++; arquitecturas de microprocessadores e arquitecturas de *software* para sistemas em tempo real.

Assim, o presente artigo pretende descrever a utilização da família do microcontrolador (μ C) MSP430 na leccionação de diferentes disciplinas. Para tal, esta comunicação envolverá os conceitos fornecidos numa disciplina base que constituirá uma **Introdução à família do μ C MSP430** aos microprocessadores e a esta família em especial,

direccionando depois para disciplinas nas áreas de **Instrumentação, Automação, e Robótica**. Entre tanto, serão também apresentados trabalhos desenvolvidos no âmbito de **Investigação & Desenvolvimento**, com a intenção de melhorar as características e as capacidades a fornecer durante a leccionação destas disciplinas.

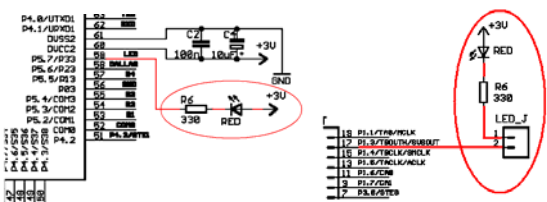
3. Introdução à família do µC MSP430

No âmbito de introduzir este tipo de tecnologias aos alunos, são leccionadas aulas que cobrem grande parte das funcionalidades do µC. Após uma apresentação das características, funcionalidades e particularidades do MSP430 e seus compiladores (no seguimento dos tópicos já aqui indicados), passa-se a uma leccionação de índole mais prática, na qual é solicitado aos alunos o desenvolvimento de diversos trabalhos: uns de ambientação aos procedimentos, expostos pelo Professor e verificados pelos alunos; e outros de projecto dirigido aos alunos cabendo-lhes realizar todo o desenvolvimento da aplicação solicitada, sempre com a tutoria do Professor.

A. Alimentação do led do kit OLIMEX

Trata-se do primeiro contacto com os kits da OLIMEX. A utilização destes kits deve-se ao facto de terem incorporado numa só placa distintos periféricos como sejam: LED, botões de pressão, LCD, campainha, ligação JTAG, entre outros. Este módulo lectivo segue os seguintes passos:

- Análise do esquema dos kits OLIMEX (Fig. 1) com o MSP430F449 (STK e STK2) para identificação do porto ao qual está ligado o LED.



a) OLIMEX MSP430 STK b) OLIMEX MSP430 STK2

Fig. 1. Esquemas dos kits OLIMEX (ligação do LED).

- Apresentação do Manual TI MSP430 [5] relativo ao capítulo à configuração de portos:
 - indicação das características dos portos;
 - identificação dos registos necessários à configuração dos portos para a aplicação em causa;
 - distinção entre configurações em hexadecimal e as mnemónicas dos ficheiros de definição.

Topic	Page
9.1 Digital I/O Introduction	9-3
9.2 Digital I/O Operation	9-3
9.3 Digital I/O Registers	9-7

Fig. 2. Configuração de portos.

- Apresentação, com base no Manual IAR Systems [6], dos procedimentos de criação e compilação dos ficheiros para a tarefa em causa:
 - introdução ao IAR Embedded WorkBench;
 - apresentação da criação de um projecto;
 - especificações da programação em linguagem C (particularidades relativamente ao C standard) para criação dos ficheiros.

Creating an application project

This chapter introduces you to the IAR Embedded Workbench™ integrated development environment. The tutorial demonstrates a typical development cycle and shows how you use the compiler and the linker to create a small application for the MSP430 microcontroller. For instance, creating a workspace, setting up a project with C source files, and compiling and linking your application.

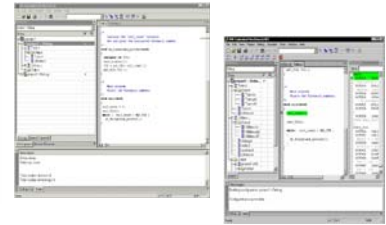


Fig. 3. Criação e compilação dos ficheiros.

- Laboratório de aplicação e execução da tarefa.

B. LCD

- Descrição dos temporizadores/contadores: *Watchdog Timer e Basic Timer*;
- Apresentação no Manual TI MSP430 do capítulo relacionado com as características dos *Watchdog Timer e Basic Timer*:
 - configuração dos registos dos temporizadores.



Fig. 4. Configuração dos registos dos temporizadores.

- Apresentação com base no Manual TI MSP430 das características do LCD:
 - descrição das características do controlador do LCD;
 - apresentação da configuração do controlador do LCD: registos de memória e de controlo.

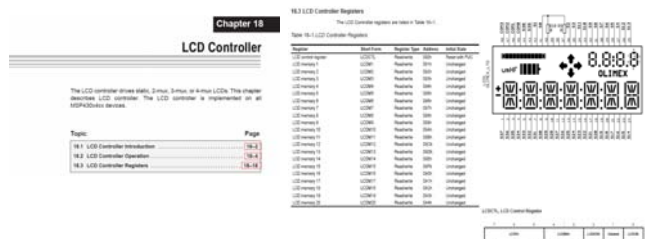


Fig. 5. Configuração dos registos do controlador do LCD.

- Laboratório de aplicação e execução da tarefa.

C. Integração: LED + botões + LCD

- Análise do esquema dos kits OLIMEX (Fig. 6) com o MSP430F449 (STK e STK2) para identificação dos portos aos quais estão ligados os botões de pressão.

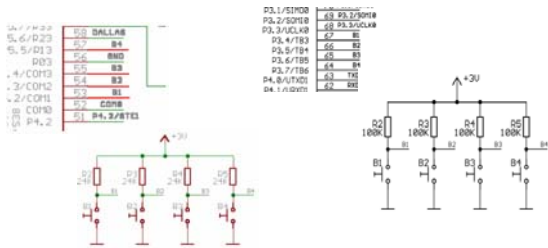


Fig. 6. Esquemas dos kits OLIMEX (ligação dos botões).

- Funcionamento por interrupções vs. *Pooling* [7]
 - análise do Manual TI MSP430 relativamente à descrição de como efectuar as interrupções aos portos, temporizadores e controlador do LCD.

Port	Register	Short Name	Address	Register Size	Initial State	
P1	Input	P1IN	0230	Read only		
	Output	P1OUT	0234	Read/write	Uninitialized	
	Direction	P1DIR	0238	Read/write	Reset with PUC	
	Interrupt Flag	P1IFG	023E	Read/write	Reset with PUC	
	Interrupt Edge Select	P1IES	0240	Read/write	Uninitialized	
	Interrupt Enable	P1IE	0244	Read/write	Reset with PUC	
	Port Select	P1SEL	0248	Read/write	Reset with PUC	
	P2	Input	P2IN	0230	Read only	
		Output	P2OUT	0234	Read/write	Uninitialized
		Direction	P2DIR	0238	Read/write	Reset with PUC
Interrupt Flag		P2IFG	023E	Read/write	Reset with PUC	
Interrupt Edge Select		P2IES	0240	Read/write	Uninitialized	
Interrupt Enable		P2IE	0244	Read/write	Reset with PUC	
Port Select		P2SEL	0248	Read/write	Reset with PUC	

Fig. 7. Registos de interrupção.

- Laboratório de aplicação e execução da tarefa.

D. Campanha

- Análise do esquema dos kits OLIMEX com o MSP430F449 (STK e STK2) para identificação do porto ao qual está ligada a campanha.
- Descrição dos temporizadores: *Timer_A* e *Timer_B*;
- Apresentação, baseada no Manual TI MSP430, das características dos temporizadores:
 - modos de operação;
 - unidades de saída e blocos de captura/comparação;
 - configuração dos registos dos temporizadores.

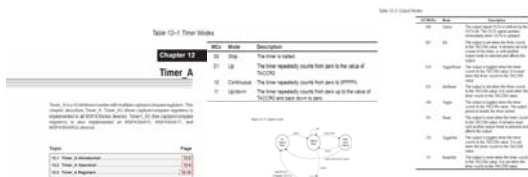


Fig. 8. Configuração dos temporizadores *Timer_A* e *Timer_B*.

- Apresentação a partir do Manual IAR Sytems da sintaxe para criação de interrupções para esta tarefa (análise também dos Manuais MSPGCC e Code Composer Essentials):
 - especificações da linguagem C para criar sinais PWM que accionem a campanha.

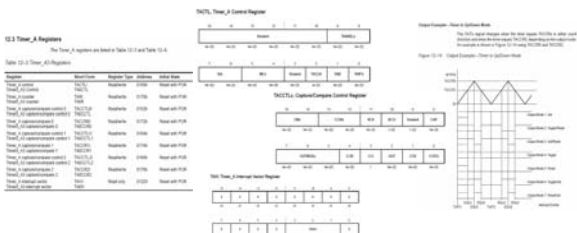


Fig. 9. Criação de interrupções para gerar sinais PWM.

- Laboratório de aplicação e execução da tarefa.

E. Relógio LCD com o kit OLIMEX

Trata-se já de uma tarefa de desenvolvimento destinada a criar um relógio que permute no LCD as configurações hh:mm e mm:ss pela pressão de botões.

- análise do esquema dos kits OLIMEX (Fig. 6) com o MSP430F449 (STK e STK2);
- configuração dos registos de portos relacionados com o LED e com os botões;
- configuração dos registos do controlador LCD;
- configuração dos temporizadores *Basic Timer* e *Watchdog Timer*;
- definição de interrupções;
- laboratório de aplicação e execução da tarefa.

4. Instrumentação

Nesta área do conhecimento foi pedido aos alunos o desenvolvimento de um dispositivo microprocessado com capacidade de detectar e armazenar eventos temporais. Esses acontecimentos seriam detectados por dois sensores diferentes: um sensor de luminosidade, e outro de vibração [8 - 11].

Sempre que o dispositivo detectasse um destes eventos deveria armazenar em memória um carimbo temporal quer da activação quer da desactivação.

Este tipo de equipamento é utilizado principalmente na realização de trabalhos de auditoria em que se pretende tomar conhecimento do historial da instalação sob observação. Para que isso seja possível o *data logger* deve ter capacidade para funcionar em permanência ao longo de vários dias.

A solução proposta pelos alunos, fazendo uso dos recursos como *Basic Timer*, LCD, USART, botões, e LED, foi a seguinte:

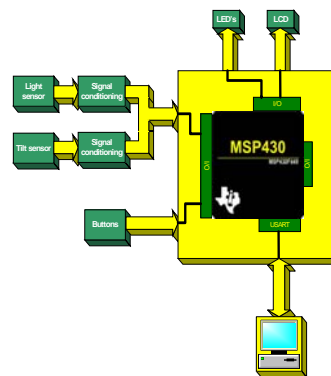


Fig. 10. Esquema de princípio.

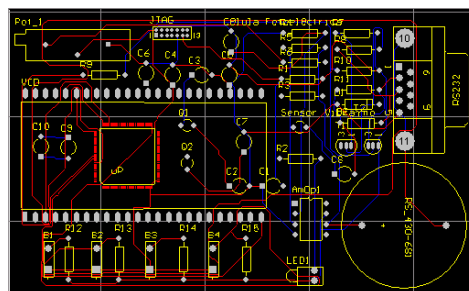


Fig. 11. Solução final.

5. Automação

Todos os sistemas de aquecimento central têm em conta a obtenção do máximo conforto com o mínimo consumo. Existem complementos para equipar a instalação que possibilitam a economia de combustível sem renunciar ao conforto total do lar. São elementos capazes de controlar a temperatura ambiente de forma a que esta se mantenha constante, independentemente do clima exterior.

Com este trabalho pretende-se que o aluno tenha um conhecimento alargado da programação de microprocessadores e de sistemas embutidos autónomos.

O sistema deve possuir as seguintes características fazendo uso dos seguintes recursos: digital I/O; controlador do LCD e conversor analógico/digital:

- relógio.
- indicação da temperatura actual.
- possibilidade de 2 mudanças diárias de nível de temperatura.
- funcionamento manual ou automático.
- tecla para consultar a temperatura de controlo programada para o período em curso.

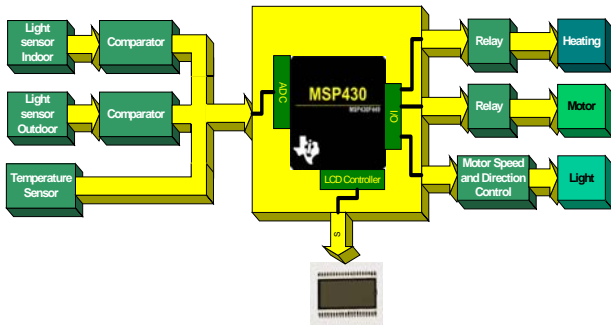


Fig .12. Esquema de princípio.

6. Robótica

Um dos trabalhos de desenvolvimento apresentado aos alunos consistiu na implementação de uma unidade de controlo de um braço mecânico FESTO [8 - 11]. Pretendia-se que possuísse as seguintes funcionalidades mínimas: ligação a PC através de porta RS232; funcionamento de todos os graus de liberdade do robot; gravação de programas pré-definidos; utilização das 16 entradas/saídas digitais; e importação de programas usando linguagem CAM.

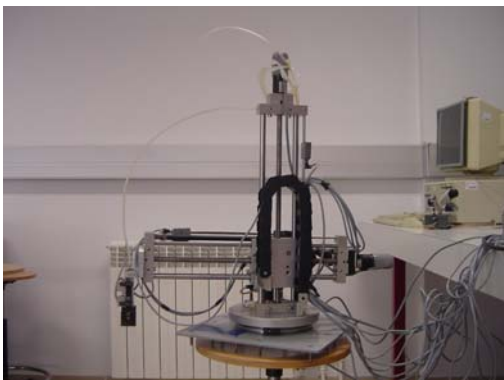


Fig .13. Braço mecânico FESTO.

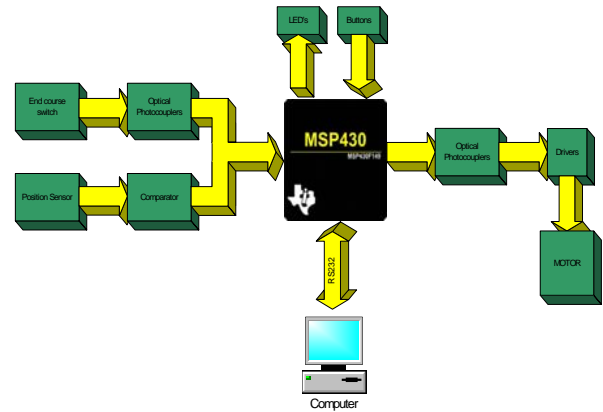


Fig .14. Esquema de princípio.



Fig .15. Solução encontrada.

Este trabalho destinou-se a desenvolver as capacidades inventivas dos alunos.

7. Investigação & Desenvolvimento

A. Kit avançado do MSP430 destinado ao ensino

Com este trabalho ambiciona-se desenvolver um *kit* em placa PCB com vários módulos conectados a um único MSP430. Um dos objectivos consiste ainda em dotar os alunos de conhecimentos sobre como desenhar sistemas baseados no MSP430.

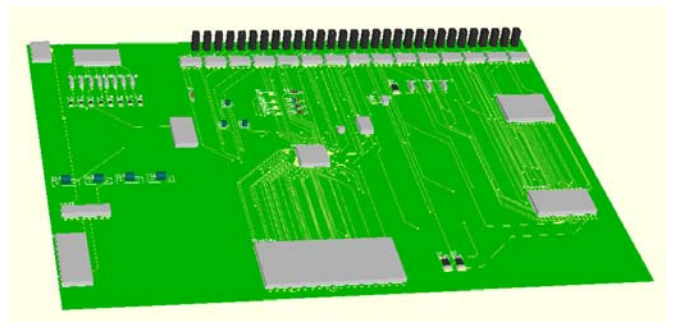


Fig .16. Protótipo do *kit*.

As funcionalidades que se pretendem com este *kit* são:

- possibilidade de ligar e desligar os módulos;
- acesso a todos os pinos do microcontrolador;

- vários LED conectados aos portos;
- teclado;
- LCD;
- I/O digitais protegidas (100 mA);
- comunicação série (RS232);
- Comunicação por *Ethernet*;
- sensores de temperatura e luminosidade.

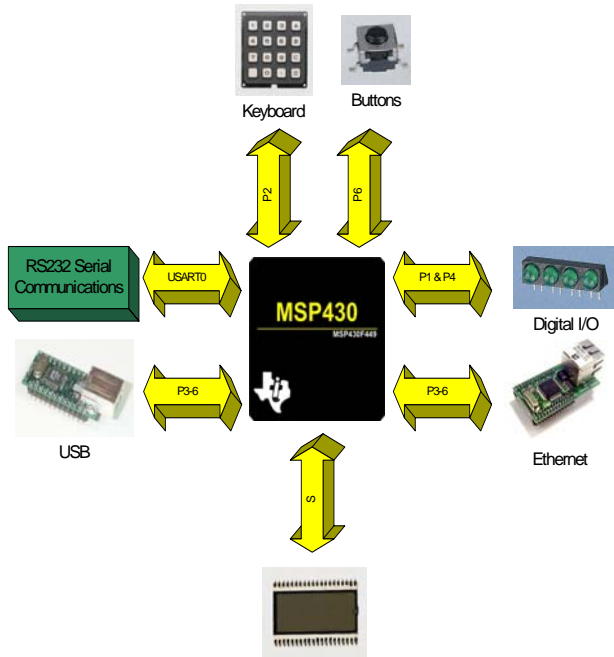


Fig. 17. Esquema de princípio.

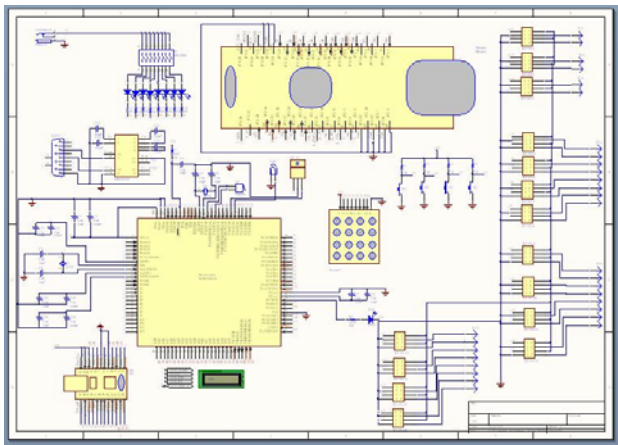


Fig. 18. Esquema de ligações.

Pretende-se com este trabalho dotar os alunos de conhecimentos com interligar vários dispositivos ao microprocessador, além do conhecimento sobre como construir placas PCB.

B. Sistemas de tempo real – Análise do SOUBI para família TMS430 [12 – 14]

As técnicas de análise da escalonabilidade de um sistema usualmente ignoram o tempo despendido pelo executivo para efectuar as mudanças de contexto entre tarefas e a manipulação das listas de tarefas pendentes, mas, em algumas situações, esta sobrecarga de CPU pode ser significativa e não deve ser negligenciada.

Na maioria dos executivos de tempo real, cabe à rotina de serviço à interrupção que gere o relógio, *tick*, (o único código que é executado na prioridade máxima do processador) a manipulação da lista de tarefas pendentes. Quando não existem tarefas nessa lista, a sobrecarga desta pode ser mínima, mas se nela existirem muitas tarefas periódicas com o mesmo tempo de activação, o *tick* pode custar centenas de microssegundos de CPU.

Para se ter em consideração o atraso na resposta das tarefas provocado pela manipulação das listas de tarefas pendentes, é necessário conhecer o funcionamento interno do executivo e ter conhecimento do valor dos seguintes parâmetros:

- C_{CLK} (sobrecarga causada pelo *tick* assumindo que não existem tarefas periódicas para manipular);
- C_{PER} (sobrecarga causada pela activação de uma tarefa pelo *tick*);
- T_{CLK} (Período do *tick*).

Um outro problema que afecta o desempenho dos sistemas baseados num kernel de tempo real é a intrínseca mudança de contexto entre tarefas (CD), ver figura 19. Este parâmetro corresponde ao tempo de processador compreendido entre o fim de uma tarefa e o início de outra.

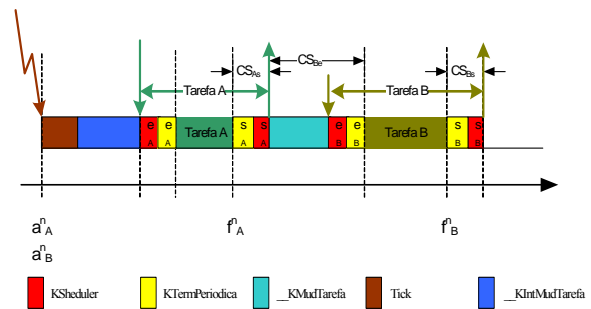


Fig. 19. Mudança de contexto no SOUBI.

Os parâmetros indicados anteriormente avaliam o desempenho do SOUBI mas também são extremamente condicionados pela arquitectura do processador e pelo seu modo de funcionamento. Para este trabalho o processador Texas MSP430F449 e está configurado da seguinte forma:

- Tick source → WDT (Modo: “*interval timer mode*”; clock source: “ACLK”; *Timer interval select*: WDT clock source / 512)
 - ACLK: *low frequency 32768 Hz watch crystals*
 - MCLK: 1048576 MHz (*Default after a PUC*)
- Nestas condições, os valores obtidos para os parâmetros invocados anteriormente são:
- $T_{CLK} = 15,6 \text{ ms}$ (figura 20)
 - $C_{CLK} = 216 \mu\text{s}$
 - $C_{PER} = 5 \times 1^{\text{a}} \text{ activação} + 35 \times (\text{N}^{\circ} \text{ activações} - 1 + \text{N}^{\circ} \text{ de tarefas}) \times 29 \mu\text{s}$
 - $CD = 274,3 \mu\text{s}$

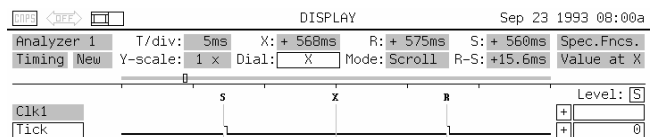


Fig. 20. Medida do período do *tick rate*

TABELA I
DADOS EXPERIMENTAIS (ANÁLISE DO TICK RATE)

	Número de Activações				
	0	1	2	3	
Número de tarefas	0	216 μ s	--	--	--
	1	245 μ s	249 μ s	--	--
	2	274 μ s	279 μ s	313 μ s	--
	3	303 μ s	308 μ s	343 μ s	378 μ s

TABELA II
DADOS EXPERIMENTAIS (ANÁLISE DA MUDANÇA DE CONTEXTO)

Kernel internals	Begin (CD)	End (CD)	TOTAL
_KIntMudTarefa	44,8 μ s	--	44,8 μ s
KScheduler	77,3 μ s	6,7 μ s	84,0 μ s
KTermPeriodica	48,8 μ s	6,7 μ s	55,5 μ s
_KMudTarefa	90,0 μ s	--	90,0 μ s

Conhecidos os parâmetros, a análise do tempo de resposta do sistema passa a ser determinado através da equação:

$$Rwc_i^{(n+1)} = C_i + CD + \sum_{k \in hp(i)} \lceil Rwc_i^{(n)} / T_k \rceil \times (C_k + CD) + \lceil Rwc_i^{(n)} / T_{CLK} \rceil \times C_{CLK} + \sum_{k \in pts} \lceil Rwc_i^{(n)} / T_k \rceil \times C_{PER}$$

8. Conclusões

Com este trabalho procurou-se dotar os alunos formados pelo DEM de conhecimentos na área do microprocessamento para aplicação em disciplinas com a automação, a instrumentação e a robótica.

O trabalho que se tem vindo a realizar através de uma abordagem global do tema, procurando envolver diversas disciplinas, começa já a dar resultados, verificados na crescente motivação dos alunos em desenvolver trabalhos que exigem este saber.

Para além da componente lectiva têm sido desenvolvidos trabalhos paralelos que visam o aproveitamento dos recursos existentes, como seja o desenvolvimento de *kits* didácticos para apoio ao ensino, ou *software* específico que permite retirar deles um melhor proveito.

Referências

- [1] www.olimex.com
- [2] www.ti.com/tmwccs
- [3] www.iar.com
- [4] <http://mbspgcc.sourceforge.net/>
- [5] Texas Instruments, User's guide – MSP430x44xx Family, Mixed Signal Products, 2004.
- [6] IAR Systems, MSP430 IAR Embedded Workbench™ IDE User Guide for Texas Instruments' MSP430 Microcontroller Family, IAR Systems, 2004.
- [7] Chris Nagy, Embedded Systems Design using the TI MSP430 Series, Embedded Technology Series, Newnes, Elsevier Science, 2003.
- [8] Aurélio Campilho, Instrumentação Electrónica – Métodos e Técnicas de Medição, FEUP Edições, 2000.
- [9] Albert D. Helfrick e William D. Cooper, Instrumentação Electrónica Moderna e Técnicas de Medição, Prentice Hall do Brasil, 1994.
- [10] G. Asch, Acquisition de données – Du capteur à l'ordinateur, Dunod, 1999.

- [11] Larry D. Jones e A. Foster Chin, Electronic Instruments and Measurements – 2nd edition, Prentice Hall – International Editions, 1991.
- [12] Tata Research Development & Design Centre, “Real-time Systems Specification, verification and Analysis”, Ed. Mathai Joseph, revised version with corrections June 2001.
- [13] M.H. Klein, J.P. Lehoczky e R. Rajkumar, “Rate-monotonic analysis for real-time industrial computing”, *IEEE Computer*, Volume 27, Issue 1, January 1994.
- [14] Bruno Ribeiro, “SOUBI – Núcleo de Sistema Operativo para Tempo Real - Manual de utilização”, 2001.