

# MSP430 MICROCONTROLLERS ESSENTIALS - A NEW APPROACH FOR THE EMBEDDED SYSTEMS COURSES: PART 1 - OVERVIEW AND TOOLS

*Pedro Dinis Gaspar, António Espírito Santo and Bruno Ribeiro*

Electromechanical Engineering Department, Engineering Faculty, University of Beira Interior  
Edifício 1 das Engenharias, Calçada do Lameiro, 6201-001 Covilhã, Portugal  
phone: + (351) 275329759, fax: + (351) 275329972, email: dinis@ubi.pt, aes@ubi.pt, bruno@ubi.pt

## ABSTRACT

*The embedded systems (ES) formation require a broader set of knowledge, abilities and skills including informatics and electronics concepts in order to develop highly creative and imaginative applications based in analytical studies. Moreover, in an effort to improve the education quality it needs to be followed with intense hands-on laboratories. This paper presents a new approach for embedded systems courses appropriate for both high school and undergraduate classrooms, that has been conceived and designed to accomplish these goals, while motivating and equipping this next generation of engineers to rise to future challenges. The course structure was defined in order to be easy to understand and provide a logical flow along the topics, as it mostly progresses from simple topics to more advanced ones. The developed materials include slides for class room teaching, explanatory documents for student and educators future reference, laboratories, tests, programs and application examples after each chapter. Each module is dedicated to a specific aspect of the MSP430 device, including the description of a range of peripherals. This is the first part of the paper presenting the outline of the course. Particularly, this paper identifies the course need, presents its structure, and the initial subjects covering an introductory overview in logic design and embedded processors and a description of the available software and hardware development tools for the MSP430.*

## 1. INTRODUCTION

Students formation in embedded systems (ES) demands for specific knowledge acquired mainly in undergraduate courses of electrical and informatics engineering classes. The practical experience of these concepts is very important as part of engineering education. Several key concepts are reinforced during lectures, as for example, the ones concerning with control systems, hardware design, data acquisition and microprocessors real time programming. Thus, all approaches for the ES education need to be simultaneously motivating and stimulating both for students and lecturers through the availability of suitable learning tools. As an example, robotics is being increasingly used as a vehicle for motivating students to learn ES, artificial intelligence, computer science, and even to introduce generic scientific and engineering concepts [1-8]. Graphical programming tools, as the one described in [9], are also used to introduce advanced concepts to the students.

In Electrical related engineering courses at the University of Beira Interior (UBI), students early contact with ES based on the MSP430 of Texas Instruments (TI). The use of this microcontroller is mostly related to its low cost, to the available number of hardware starter kits and software integrated development environment (IDE), either free or with a reduced cost. Other aspects justify device usage are, among others, the amount of peripherals integrated available, the low power features, the high pin out counting, the C/C++ or assembly programming, the rapid learning curve. The TI MSP430 family are low power processing devices, especially developed for low-power applications. Due to its simple and actual structure, it has been increasingly a choice for teaching ES on many educational institutions around the world. To overcome the evident limitations from the low processing capabilities, it are equipped with a wide range, and extremely powerful, of peripherals, that for users with little experience in this field, can present a complex configuration/operation.

One successful example of using the MSP30 consisted in the substitution of the RoboSapien control/regulation electronics by the MSP430 to replicate its operation [10]. The main goal of this project was to motivate the students, while at the same time to develop a technology demo vehicle.

Our main goal on redefining the course structure is to gather the ES topics, both software (SW) and hardware (HW), that can be found disperse in literature [11-21], which covers: assembly/C language of various processors; MCU devices; technologies; and even PCB soldering/design, making use of different SW/HW platforms, as presented in Figure 1.

ES education is difficult to generalize because: includes a broad range of concepts from many disciplines; is used in a large range of applications; includes many different technologies; is subjected to a wide range of design constraints; at the same time many different ES architectures and platforms are in use today. Evaluation of the suitability of an ES architecture as a general-purpose laboratory platform is produced in [22]. Real-world project-based learning strategy as a pedagogical tool for ES education integrating and formulating the multi-disciplinary knowledge previous learned can be found in [23]. Research of on-campus process of practical training with experiences suitable for IT companies is described in [24].

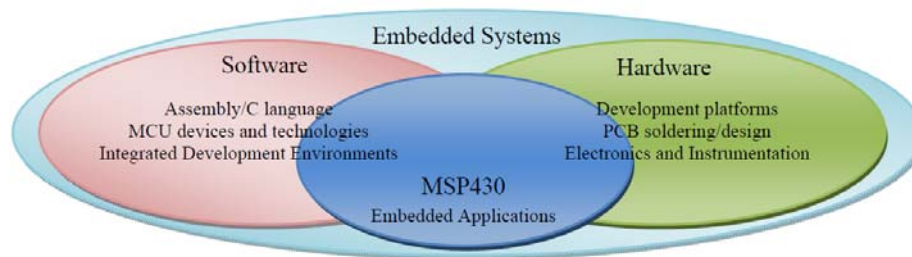


Figure 1 – Course structure philosophy gathering several Embedded System topics.

Novel schedule of ES education with professional engineering practice and progressive experiential is described in [25]. An educational method using student experiments, providing a deeper knowledge on system modelling, SW/HW trade-off and development of SW/HW modules is produced in [26]. Iterative and incremental methodology dividing the ES curriculum in two major parts: ES guidance courses and programming skills training are described in [27]. Development of a scalable MCU peripheral system (Micro-eBlocks) to facilitate the student's learning of ES based modules is described in [28]. These modules are a set of plug-and-play input-output, sensors, communication and processor modules, which can be used to address a wide variety of ES application need; ES education strategy focused on lecture–laboratory integration and laboratory learning [29] using a learning model that captures lower- and higher-order cognition levels. The learning experience in the laboratory is characterized using a technique to assess cognitive behaviour; wireless ES teaching methodology and laboratory setup design allowing the students to apply their previous knowledge and to experience actual systems engineering by designing and implementing a large-scale team project [30].

Based in their experience teaching signatures involving microcontrollers, the authors in partnership with Texas Instruments, developed a set of teaching vital core materials to enable educators and academics to teach ES, using devices from the Texas Instruments MSP430 family [34]. The materials produced include slides for class room teaching, explanatory documents for future references both for students and educators, laboratories, tests, programs and application examples after each chapter and then explanation going through them helps understand them easily. The tutorials can be used as student guides to a series of modules and laboratory exercises. Each module is dedicated to a specific facet of the device, including the description of a range of peripherals. These materials also include the step-by-step project development using different software development and the description of the hardware starter kits tools (MSP430FG4618/F2013 Experimenter's board and the eZ430-F2013 and eZ430-RF2500 MSP430 USB Stick Development Tools) available to perform the laboratory exercises.

This is the first paper of three presenting the outline of the course. Particularly, this paper identifies the need for such a

course, presents its structure, and the initial subjects covering an introductory overview in logic design and embedded processors and a description of the available software and hardware development tools for the MSP430.

## 2. COURSE STRUCTURE

The course structure was defined in order to be easy to understand and provide a logical flow along the topics, as it mostly progresses from simple topics to more advanced ones. Although it was primary developed to support teaching activities, it intends also to be helpful to existing embedded system designers or to those people new to ES such as college students, high school and middle school students, and other hobbyists who wish to make their own projects enabling them to learn about how to use efficiently in short time a MSP430 family device.

We perceive an increasing trend of tremendous amount of crossover in the teaching content and projects of all engineering disciplines (especially with Mechanical and Biomedical), and this creates a huge gap for professors who need to add electronic devices (such as robotics or some embedded control) to their courses, yet do not have the time or background to go into the complexity necessary for learning an entire hardware platform inside and out. This material would be extremely useful to programs looking to integrate some laboratories or modern projects into other engineering curriculum such as those that were mentioned. Also, this course reference additional documents on TI website like [35-39] to provide added in-depth information. Each section contains a topic devoted to laboratory exercises. As the course intends to be helpful also to those people that does not has any background on the applications development for this microcontroller, the code is already given, being only asked to the student to fill in the blank code devoted to the configuration of the specific peripheral. However and for pedagogy reasons, is more challenging for senior students to state only the requirements and let them figure how to code. For these advanced students, a blank slate will force them to think how to structure the program code and what function/block needs to setup, instead of think for themselves how to program the code that will perform the task. At the end of each section are included Quiz and FAQ sections to provide a insight and a self evaluation of the main topics presented in the section completed with some questions and difficulties that usually arise using the MCU peripherals.

## 2.1 Introductory Overview

We realize that some of the introductory material in this section might not be used in that environment given the student's background, if we consider that most of seniors courses would already have taken introductory courses in logic design and embedded processors. Nevertheless, this content can be seen as a refresher of a mixture of basic knowledge and general information of MCU architecture and C and Assembly programming languages based on [40-42]. So, the main goal of this section is to review fundamental aspects of analogue and digital signals and systems, digital codes, the binary, decimal and hexadecimal numbering systems and conversions between them. First, it briefly describes the differences between floating-point and fixed-point processor cores, their numerical representation based on the IEEE 754 standard, and looks at the effect on performing mathematical operations and clock cycles used. It also has a sub-section for beginners devoted on how to read technical specifications in MSP430 datasheets. The next section presents a general overview on C programming language, starting from programming styles, data declaration (C data types, modifiers, and identifiers), operators and expressions used by the MSP430 to perform logical and mathematical operations used by the MSP430 to perform logical and mathematical operations. The section ends with some specific topics related with good software practices and how to take advantage of the low power capabilities of the MSP430. Probably, for beginners, this last section only will be completely understood after acquiring some background knowledge which comes on later chapters, however the authors intention is to instil from the beginning the concept and the correct usage of the low power features of this microcontroller.

## 2.2 Software Development Tools

This section presents the main characteristics of the MSP430 Integrated Development Environments (IDEs),

both from TI and from third parties. Due to the wide range of hardware platforms available, special attention is given to Code Composer Essentials (TI) [43-45] and Embedded Workbench (IAR Systems) [46] IDEs. These development tools have advanced capabilities to support the development of applications for the MSP430 family. Among them are the support for the use of breakpoints, either hardware or software. The IDEs support code debugging activities, with support for features such as code step-by-step execution, or fast and efficient access to registers and memory locations. There is complete compatibility between the C programming language syntax used and the great diversity of code examples available. Topics covered in this section are the structure and management (source files, compiling, assembling and linking operations) of projects developed in both C (mainly) and/or Assembly language. The basic functions and step-by-step project development are given for each tool. During the development of the laboratory examples, the software development tools will be used to make use of the microcontroller features. These IDEs allow applications to be written, compiled, assembled, linked, debugged and run on MSP430 hardware. The examples for application debug, using breakpoints, advanced break points with triggers, memory and usage registers, are provided with screen-snapshots in order to ease the understanding as it is shown in Figure 2. Although the TI's IDE is a powerful tool, the IAR Embedded Workbench includes a simulator that gives the possibility for students to program, simulate and train at home additional features such as in the debug window each student sees directly what the source code generates; the LCD simulation shows immediately what happens and changing register contents gives immediate reaction. At the end of the chapter are described some additional third parties that have developed Software Development Tools to configure and program the MSP430 hardware development tools.

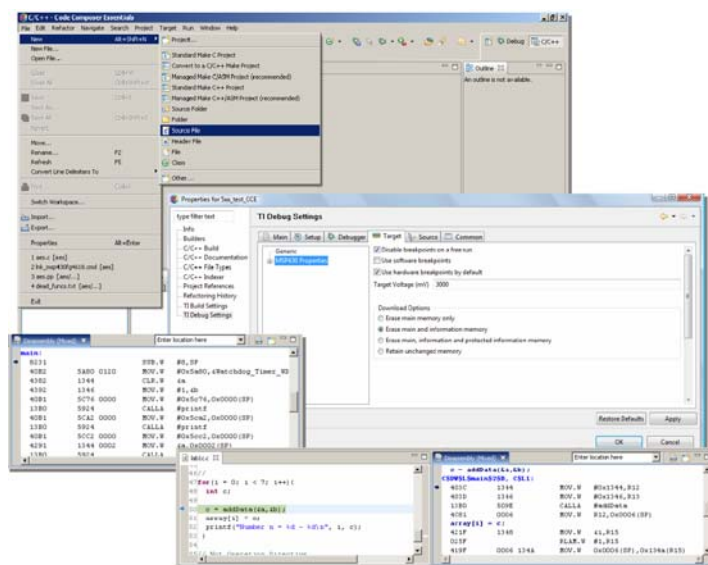


Figure 2 – CCE workbench windows: C/C++ perspective; Device options configuration; Disassembly; code.

### 2.3 Hardware Development Tools

The aim of this section is to describe the MSP430 hardware development tools. During the sections that follow, the laboratories will be supported by either one of two platforms. Depending on the specific peripheral that is studied, it uses the eZ430-F2013/eZ430-RF2500 MSP430 USB Stick Development Tool and/or the MSP430FG4618/F2013 Experimenter's board shown in Figure 3. The main features and integrated peripherals for each platform are described. The eZ430-F2013 [47] is a MSP430 development tool used to evaluate the 'F2013 and develop projects on hardware that is contained on a USB stick. The USB port provides enough power to operate the MSP430 without requiring an external power supply. All 14-pins on the MSP430F2013 are accessible on the MSP-EZ430D target board for easy debugging and interfacing to peripherals. One of these digital I/O ports is connected to an LED for visual feedback. The eZ430-RF2500 [48] is a USB-based MSP430 wireless development tool used to evaluate the MSP430F2274 microcontroller and CC2500 2.4-GHz wireless transceiver. The eZ430-RF2500T target board is an out-of-the box wireless system that may be used with the USB debugging interface, as a stand-alone system, with or without external sensors, or may be incorporated into an existing user design. The MSP430 experimenter's board [49] features a MSP430F2013 and a MSP430FG4618, and is compatible with TI's wireless evaluation modules. The combination of these two MCUs provides nearly every peripheral available in the MSP430 family. The integrated TI wireless evaluation module header and the large amounts of RAM on the MSP430FG4618 makes it an ideal platform for wireless applications. Some information is also provided covering the new hardware development tool (MSP-EXP430F5438 experimenter's board [50]) that includes a MSP430x5xx device. The device features and hardware installation of the flash emulation tool (MSP-FET430 [51-52]) are presented also. It allows the application development on the MSP430 MCU. Two debugging interfaces are available, USB and parallel port. These two debugging interfaces are used to program and debug the MSP430 in-system through the JTAG interface or the pin saving Spy Bi-Wire (2-wire JTAG) protocol. The hardware development tools provided

by third parties are listed in the last topic of the section, since the usage of a hardware kit depends on the specific needs of the user, benefiting the use of some peripherals at the detriment of others.

### 3. CONCLUSIONS

It was presented the teaching structure in the lectures that use microprocessors at the undergraduate and graduate Electrical related courses at the University of Beira Interior, Portugal that make use of microcontrollers, including signatures such as, Instrumentation and Measurements; Data Acquisition; Automation and Robotics; Industrial Informatics; Real Time Systems; Embedded Systems; Bionic Systems and Monitoring and therapy medical portable devices. The former last five signatures integrate several knowledge that are acquired during the courses. This paper presents the outline of the course. Particularly, the course structure, and the initial subjects covering an introductory overview in logic design and embedded processors and a description of the available software and hardware development tools for the MSP430. The projects development making use of the microcontroller has proven to be a valuable teaching tool for motivating and stimulating the students, allowing the reinforcement of several key concepts discussed in undergraduate signatures. With this pedagogical approach, the students gain much more experience since they are challenged to develop, not overwhelming, but much more complex projects, while keeping them motivated.

### ACKNOWLEDGMENTS

The authors thank the support given by Texas Instruments; and particularly the help provided by Robert Owen (TI University Programme Manager).

### REFERENCES

- [1] I. Horswill, "A laboratory course in behavior-based robotics," *Intelligent Systems and Their Applications*, IEEE, 15(6): Nov.-Dec. 2000.
- [2] M. Rosenblatt and H. Choset, "Designing and implementing hands-on robotics labs," *Intelligent Systems and Their Applications*, IEEE, 15(6): Nov.-Dec. 2000.



Figure 3 – Hardware development tools.

- [3] J. A. Piepmeier, B. E. Bishop and K. A. Knowles, "Modern robotics engineering instruction," *Robotics & Automation Magazine*, IEEE, 10(2): June 2003.
- [4] B. A. Maxwell and L. A. Meeden, "Integrating robotics research with undergraduate education," *Intelligent Systems and Their Applications*, IEEE, 15(6): Nov.-Dec. 2000.
- [5] R. Kuc, E. W. Jackson and A. Kuc, "Teaching introductory autonomous robotics with JavaScript simulations and actual robots," *IEEE Trans. Education* 47(1): Feb. 2004.
- [6] J. M. Conrad, "Stiquito for robotics and embedded systems education," *Computer* 38(6): 2005.
- [7] K. S. Rawat and G. H. Massiha, "A hands-on laboratory based approach to undergraduate robotics education," in *Proc. ICRA '04. 2004 IEEE International Conference on Robotics and Automation*, vol. 2, Apr-May, 2004.
- [8] K. Nagai, "Learning while doing: practical robotics education," *Robotics & Automation Magazine*, IEEE, 8(2): June 2001.
- [9] J. B. Maloney, L. Kafai, N. Y. Rusk, B. Silverman and M. Resnick, "Scratch: a sneak preview," in *Proc. 2<sup>nd</sup> Int. Conf. on Creating, Connecting and Collaborating through Computing*, 2004.
- [10] P. D. Gaspar, A. E. Santo, B. Ribeiro and H. Santos, "Robosapien powered by TI MSP430," in *Proc. Texas Instruments V Advanced Technical Conference (ATC 06)*, Munich, 2006.
- [11] J. Catsoulis, *Designing embedded hardware*. O'Reilly Media, May 2005.
- [12] S. Heath, *Embedded systems design, 2<sup>nd</sup> Ed*. Elsevier Science, Newnes, 2002.
- [13] M. Barr, *Programming embedded systems: with C and GNU development tools, 2<sup>nd</sup> Ed*. O'Reilly Media, Oct. 2006.
- [14] Z. Karakehayov, K. S. Christensen and O. Winther, *Embedded systems design with 8051 microcontrollers: Hardware and software (Electrical and Computer Engineering)*. CRC Press, 1999.
- [15] T. Noergaard, *Embedded systems architecture: A comprehensive guide for engineers and programmers (Embedded Technology)*. Elsevier Science, Newnes, 2005.
- [16] J. Ganssle, T. Noergaard, F. Eady and L. Edwards, *Embedded hardware (Newnes Know It All)*. Elsevier Science, Newnes, 2007.
- [17] J. H. Davies, *MSP430 Microcontroller basics*. Elsevier Science, 2008.
- [18] K. Hyder and B. Perrin, *Embedded systems design using the Rabbit 3000 microprocessor: Interfacing, networking, and application development (Embedded Technology)*. Elsevier Science, 2004.
- [19] J. S. Parab, S. A. Shinde, V. G. Shelake and R. K. Kamat, *Practical aspects of embedded system design using microcontrollers*. Springer Netherlands, 2008.
- [20] J. Ganssle *The art of designing embedded systems, 2<sup>nd</sup> Ed*. Elsevier Science, Newnes, 2008.
- [21] C. Nagy, *Embedded systems design using the TI MSP430 series*. Elsevier Science, Embedded Technology Series, 2003.
- [22] K. G. Ricks and D. J. Jackson, "A case for the VMEbus architecture in embedded systems education," *Transactions on IEEE*: 49(3) 2006.
- [23] R. C. Hsu and W.-C. Liu, "Project based learning as a pedagogical tool for embedded system education," in *Proc. ITRE 2005. 3<sup>rd</sup> International Conference on Information Technology: Research and Education*. June 2005.
- [24] H. Dai, Z. Jia, X. Li and Y. Guo, "Practical training in the embedded system education: A new way to narrow the gap with industry," in *Proc. ICYCS 2008. The 9<sup>th</sup> International Conference for Young Computer Scientists*. Nov 2008.
- [25] L. Xiaojuan, G. Yong and Y. Huimei, "Curriculum development and progressive engineering practice design in embed system education", in *Proc. MESA 2008. IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications*. Oct. 2008.
- [26] H. Mitsui, H. Kambe, D. Tilwaldi and H. Koizumi, "A student experiment method for embedded system education based on incremental upgrade," in *Proc. ICPPW 2007. International Conference on Parallel Processing Workshops*. Sep. 2007.
- [27] J. Chen, H.-M. Su and J.-H. Liu, "A curriculum design on embedded system education for first-year graduate students," in *Proc. 2007 International Conference on Parallel and Distributed Systems*, Volume 2. Dec. 2007.
- [28] C.-C. Tan; S.-L. Tan, E.-S. Chng and W.-B. Goh, "MICRO-EBLOCK: A modular platform for embedded system education," in *Proc. 2008 International Conference on Computer Science and Software Engineering*, Volume 5. Dec. 2008.
- [29] D. T. Rover, R. A. Mercado, Z. Zhang, M. C. Shelley and D. S. Helvick, "Reflections on teaching and learning in an advanced undergraduate course in embedded systems," *IEEE Transactions on Education* 51(3): 2008.
- [30] J.-S. Chenard, Z. Zilic and M. Prokic, "A laboratory setup and teaching methodology for wireless and mobile embedded systems," *IEEE Transactions on Education* 51(3):2008.
- [31] P. D. Gaspar, A. E. Santo, H. Santos and B. Ribeiro, "Nova abordagem para o ensino da robótica baseada na modificação do RoboSapien," in *Proc. of Engenharia '2007 - Inovação e Desenvolvimento*, Universidade da Beira Interior, Covilhã - Portugal, Nov., 2007.
- [32] H. Santos, P. D. Gaspar, A. E. Santo and B. Ribeiro, "Ensino de microcontroladores usando a família MSP430," in *Proc. of Engenharia '2005 - Inovação e Desenvolvimento*, Universidade da Beira Interior, Covilhã - Portugal, Novembro, 2005.
- [33] A. Guerman, H. Santos, P. D. Gaspar, A. E. Santo and C. Santos, "Módulo de gestão e resolução de exercícios: tutoria on-line,# in *Proc. of eLearning no Ensino Superior (eLES 04 )*, Aveiro, Oct, 2004.
- [34] P. D. Gaspar, A. E. Santo, B. Ribeiro and H. Santos, *Microcontrollers essentials - MSP430 Teaching ROM*. Texas Instruments, 2009.
- [35] B. Baker, "A Glossary of Analog-to-Digital specifications and performance characteristics," *Application Report SBAA147A, Texas Instruments*, 2008.
- [36] H. Grewal, "Oversampling the ADC12 for higher resolution," *Application Report SLAA323, Texas Instruments*, 2006.
- [37] J. Patterson, "A flash monitor for the MSP430," *Application Report SLAA341, Texas Instruments*, November, 2006.
- [38] K. Quiring, "Implementing an ultralow-power thermostat with slope A/D conversion," *Application Report SLAA129B, Texas Instruments*, 2006.
- [39] M. Koesler and W. Lutsch, "Programming a flash-based MSP430 using the JTAG Interface," *Application Report SLAA149D, Texas Instruments*, 2008.
- [40] B. Razavi, *Fundamentals of microelectronics*. John Wiley & Sons, 2008.
- [41] A. V. Oppenheim, A. S. Willsky and H. Nawab, *Signals & Systems, 2<sup>nd</sup> Ed*. Prentice Hall, 1997.
- [42] K. Quiring, "MSP430 Software coding techniques," *Application Report SLAA294A, Texas Instruments*, 2006.
- [43] TI, *MSP430 Optimizing C/C++ Compiler v 3.1 User's Guide*. Application Report SLAU132C, Texas Instruments (TI), 2008.
- [44] TI, *MSP430 Assembly Language Tools v 3.0 User's Guide*. Application Report SLAU131B, Texas Instruments (TI), 2008.
- [45] D. Sale, "Advanced debug capabilities of CCE," in *Proc. ATC 2008 - MSP430 Advanced Technical Conference*, Texas Instruments, Sonthofen, Germany, 2008.
- [46] IAR Systems. *MSP430 IAR Embedded Workbench™ IDE User Guide for Texas Instruments MSP430 Microcontroller Family*. IAR Systems, 2004.
- [47] TI, *eZ430-F2013 Development Tool User's Guide*. SLAU176B, Texas Instruments (TI), 2006.
- [48] TI, *eZ430-RF2500 Development Tool User's Guide*. SLAU227C, Texas Instruments (TI), June 2008.
- [49] TI, *MSP430FG4618/F2013 Experimenter's Board User's Guide*. SLAU213A, Texas Instruments (TI), Oct. 2007.
- [50] TI, *MSP-EXP430F5438 Experimenter Board User's Guide*. SLAU263A, Texas Instruments (TI), April 2009.
- [51] TI, *MSP-FET430 Flash Emulation Tool (FET) (for use with Code Composer Essentials for MSP430 Version 3) User's Guide*. SLAU157G, Texas Instruments (TI), May 2008.
- [52] TI, *MSP-FET430 Flash Emulation Tool (FET) (for use with IAR Embedded Workbench Version 3+) User's Guide*. SLAU138K, Texas Instruments (TI), May 2008.