

AUTOMATED MEASUREMENT SETUP FOR MICROCONTROLLERS AND MAGNETIC SENSORS

Răzvan DOCHIA¹, Daniela BOGDAN², Corneliu BURILEANU³

În această lucrare se vor prezenta rezultatele obținute în urma dezvoltării unor programe de automatizare a măsurătorilor. Acestea au fost folosite pentru demonstrarea tehnicilor legate de consumul de putere. Pentru validarea dar și pentru demonstrarea versatilității tehnicilor folosite vor fi prezentate și rezultatele obținute în cazul măsurătorilor unui senzor magnetic pentru care se realizează și o corelare cu rezultatele simulărilor.

In this paper there will be presented the results obtained after the implementation of an automated measurement setup. These were used to prove the proposed methods of power consumption. To validate and to demonstrate the versatility of the techniques used, there will also be a presentation of the results obtained for measuring a magnetic sensor for which a correlation with the simulation results is also performed.

Keywords: automation, power estimation, microcontroller simulator, magnetic sensor, ageing measurements

1. Introduction

Automatic testing has been a widely discussed topic and desired goal for several years in the electronics and software industries. Despite this, there is not one test engine, test management tool, or even test standard that meets all the needs, or desires, of every organization that might use it.

Automatic test should do a lot more than just ask questions of a tester, but it most certainly will not be able to infer everything about what to do in order to perform a test.

Now there are a lot of tests that need to be performed on chips before beginning mass production or even for small lots. Offering a great quantity of information in a short time is of great importance. There is software that can be

¹ PhD student, Faculty of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: razvan.dochia@gmail.com

² PhD student, Faculty of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: daniela.bogdan@yahoo.com

³ PhD student, Faculty of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: cburileanu@messnet.pub.ro

used to perform measurements but depending of application it can be more difficult to adjust the software for the desired setup. Also automatic testing means not only the measurement but also collecting and processing data. This means plotting charts, calculating averages, statistical measurements and comparing obtained results with the expected ones.

Automation is a highly effective method for reducing or substantially eliminating human error and also for time reducing. For example, the collection of data using the computer has greatly improved the reliability, in comparison to the manual methods. Routine, repetitive tasks are particularly good candidates for automation. In this category can be included also device characterization. [1]

The characterization process of the microcontrollers is a very time consuming procedure and implies detailed information about the microcontroller architecture. For energy estimations of different architectures during the design process of embedded systems an automated characterization is absolutely essential. For that matter we have implemented an automated characterization system that is also capable of performing device characterization.

2. The setup and software

The first step in developing the automation setup was the selection of the most suitable programming language in which all the desired characteristics to be implemented. In order to choose the most suitable program environment we analyzed the list of parameters to be measured and this were: current, voltage and power. Then another very important aspect was that the programming language should permit the configuration of the microcontroller and all the other equipments. The final aspect was the fact that the results should have a certain output format that can be easily use and interpret.

There are many software solutions available for the development of computer programs, which can control the equipment, like LabVIEW (created by National Instruments) and VEE (created by Agilent Technologies). There is also the possibility of the direct control with Microsoft Visual Basic or Sun Microsystems for Java, which provide a good flexibility when it is needed to work with different libraries or databases.

The language we finally choose to implement the software was Perl because we wanted to be able to include more characteristics like the power information.

Perl is a high-level programming language that uses an interpreter to translate the source code into an executable format at run time instead of being pre-compiled. First of all, this language has been selected for our environment because we wanted to use a language that has a great level of acceptance all over

the world and according to a global survey performed by Evans Data in 2008 [2] it is one of the top 3 scripting languages used worldwide.

Another reason in favor of choosing Perl is that it can be used on all platforms and it is more portable than C language. Other advantages that recommend Perl for our automatic test environment are that a huge collection of Perl modules is free (either GNU General Public License or Artistic License), it is very efficient in text and string manipulation and that dynamic memory allocation is very easy to be accomplished. Besides all these qualities, another important one is that all the scripts that are written in Perl can be accessed and integrated easily in LabVIEW; so if a user will decide in the future that it is better to switch to LabVIEW software, then all the tests that were written in Perl can be re-used in this software environment also.

The developed software created was named *Characterization & Extractor Environment (CEE)*. After the software was finished a Graphic User Interface (GUI) was developed. To fulfill our task many things were taken into account because of the need to create a complete and universal measurement tool. Keeping in mind the simplicity more options were implemented in GUI like:

- a slider to select if the test is performed on a microcontroller or on magnetic sensors (in this paper the functionality was tested both on a microcontroller and on a magnetic sensor);
- test that will be run by the microcontroller;
- frequency of running code in the microcontroller;
- voltage at which the microcontroller or the magnetic sensor will be supplied with;
- value that will be written in data memory and
- temperature information.

With all this information the CEE tool will program the microcontroller, setup all the GPIB needed devices and take the power measurements.

As the CEE tool is designated to be an automated test engine and thus reduce the human intervention, the interface of the setup offers to the user the possibility of selecting all the possible parameters or just the combination that is desired.

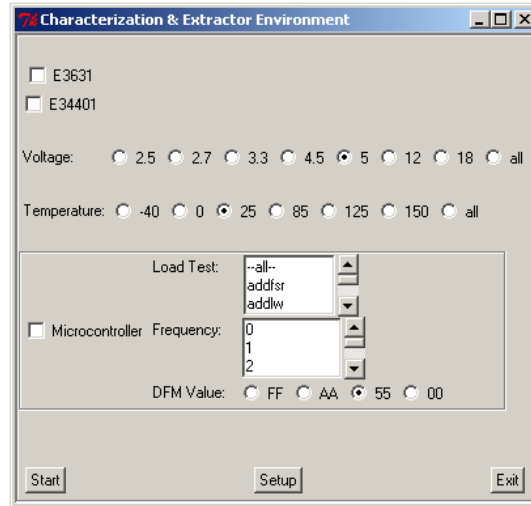


Fig. 1. User interface

3. Measurement hardware

For testing the CEE tool we chose two different devices: a microcontroller and a magnetic sensor. For the microcontroller we decided to use an 8-bit microcontroller with the enhanced mid-range core from Microchip [3]. From this family of Microchip microcontrollers we selected PIC16F1933 [4] because it is rich in peripheral modules and also has a low pin count.

The measurement setup was designed to be able to perform the measurement on the two products. The final setup included one power supply, an Agilent E3631A, and in series a multimeter, an Agilent 34401A (Fig. 2)[5].

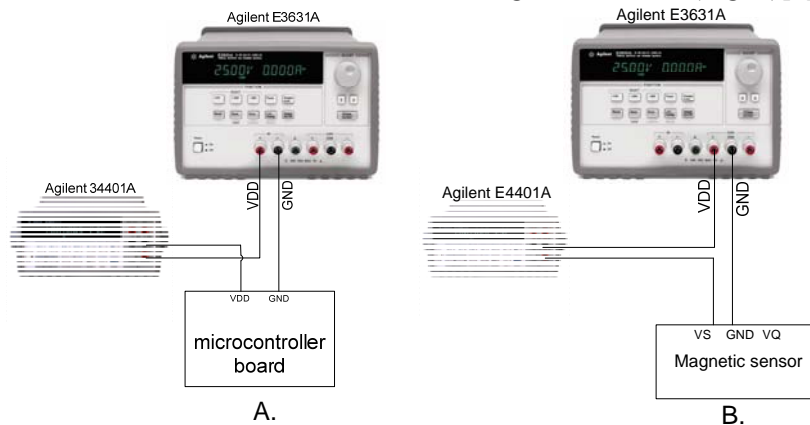


Fig. 2. The Measurement Setup for the
A. Power Extractor tool for microcontroller board; B. Characterisation of magnetic sensor

One reason we have chosen these equipments was the fact that they can be easily controlled by GPIB and they show accuracy. The same setup was used for measurement of a magnetic sensor. The measurement results were compared with the theoretical model to check if they are in the expected range. The automation helped a lot and reduced time of measurement and also human intervention in the measurements.

4. Results

4.1 Results on the microcontroller

In the 1990s, Tiwari et al [6] introduced a methodology for software power estimation at instruction level based on physical measurements. It is based on an empirical method for characterizing the power dissipation of short instruction sequences and for using these results to estimate the power/energy dissipation of a program. This methodology can be split into two parts: the characterization step and the estimation step.

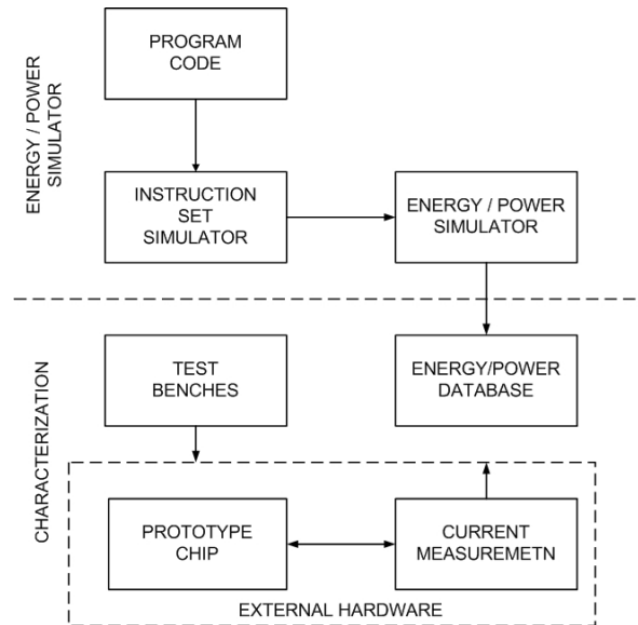


Fig.. 3. Microcontroller characterization and power estimation overview

As can be seen from Fig. 3, in the characterization step, several programs are loaded into the microcontroller. Using these so-called test benches the microcontroller is brought into certain states. An important aspect in the characterization process is the selection of instruction sequences that will be ran on the microcontroller. At the limit, it is necessary to determine the base cost of individual instructions. The base cost of an instruction may be thought as the cost

associated with the basic processing action needed to execute the instruction. Executing programs containing a loop of several instances of the given instruction will determine the instruction base cost. Power and energy estimates for the instruction are calculated from the average current draw, the supply voltage, and the number of execution cycles per instruction. Given an instruction sequence, using only the base costs of the individual instructions to model the cost of the sequence tends to underestimate the cost. This is so because costs do not take into account the effect of the circuit state change when two different instructions are executed consecutively. These costs are referred to as circuit state effects and they need to be also measured. Afterward, these cost factors are stored into an energy/power database.

The second step in the process for software power estimation at instruction level is the estimation step. In this step a program is executed on a simulator, which is providing an estimate of the power consumed by the microcontroller while executing the algorithm. Using the power information obtained from the CEE, we have created a simulator [7], which is providing an estimate of the power consumed by the microcontroller when executing the algorithm. Using this simulator we have realized comparisons between the consumption estimation of the model and physical measurements on the real platform. The program chosen for these comparisons consists of loading the accumulator of the microcontroller with values from the registers in an infinite loop.

Results of model estimations and physical measurements are presented in Table 1. All the presented results were obtained at 4.5V and had constant data written in the data memory of the microcontroller.

Table 1

Simulator results

Frequency [kHz]	Measured value [mW]	Simulated value [mW]	Error [%]
31.25	1.48	1.5	1.34
62.5	1.49	1.51	1.27
125	1.51	1.53	1.25
250	1.53	1.56	1.52
500	1.59	1.61	1.3
1000	2.58	2.59	0.58
2000	2.84	2.85	0.45
4000	3.36	3.37	0.33
8000	4.38	4.38	0.01
16000	6.24	6.23	0.22

4.2 Magnetic sensor measurements

For the magnetic sensor measurement it was selected a parameter, which was easily measured with the same, automated setup. In this case, a voltage-current measurement was used, the measurement of the current on the supply pin (Vs). First there were performed simulations on the macro-model and manual measurements and after that the automated test was run.

First the setup and results of the macromodel of the magnetic sensor can be observed in Fig. 4.

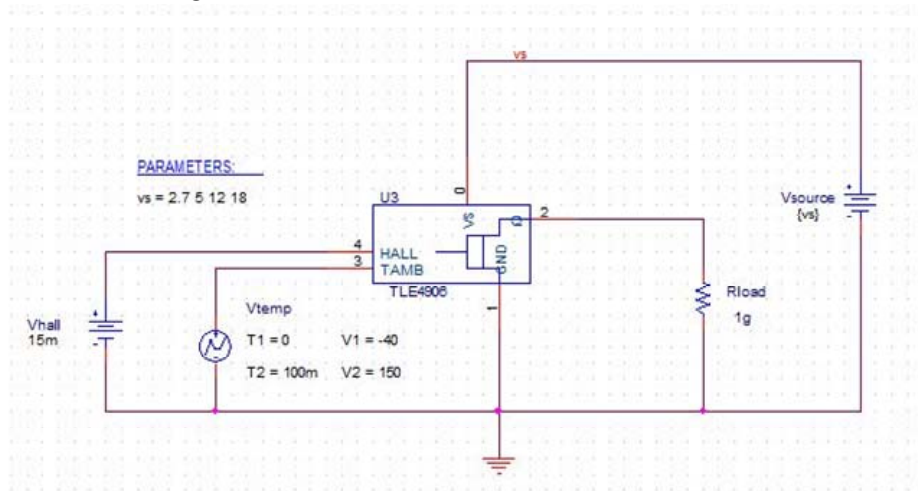


Fig.

4. The simulation setup

The sensor has three pins: one for the supply, one output and one ground. As it can be observed in Fig. 4, there is a supply source on the Vs (supply pin). The current measurements are done on the same pin.

For modeling the temperature, a voltage source has been used. This is due to the limitations of the Spice simulator. The temperature-voltage correspondence is 1:1 for the ease of simulation setup. So setting a 25V for the voltage source on the setup corresponds to a 25°C modeled temperature (the source connected in Fig. 4 to TAMB pin). This is why it was possible to use a voltage source for temperature modeling.[8].

The simulations were performed at four supply voltages: 2.7V, 5V, 12V, 18V and at a range of temperatures between -40°C to 150°C.

The actual measurement made in order to check the functionality of CEE were performed on 10 samples at three temperatures (-40°C, 25°C, 150°C) and at four supply voltages (2.7V, 5V, 12V, 18V). The current was measured on the same pin.

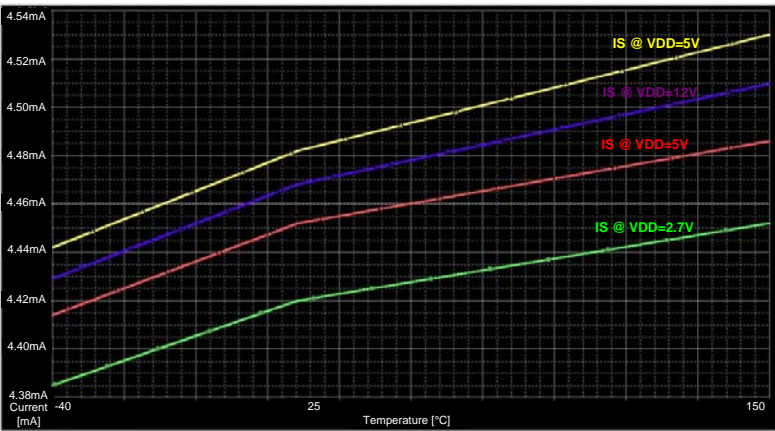


Fig. 5. The simulation results

The results after executing the automated setup can be observed on Fig. 6 to Fig 8.

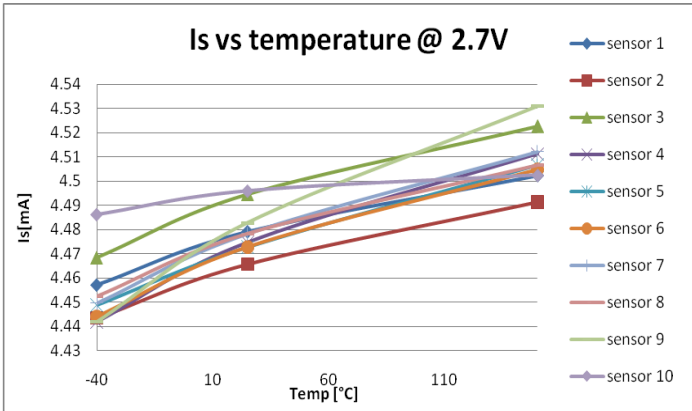


Fig. 6. Measurement results for 2.7V

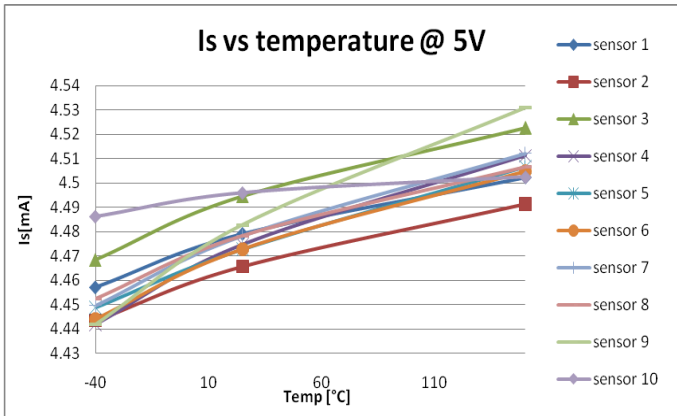


Fig. 7. Measurement results for 5V

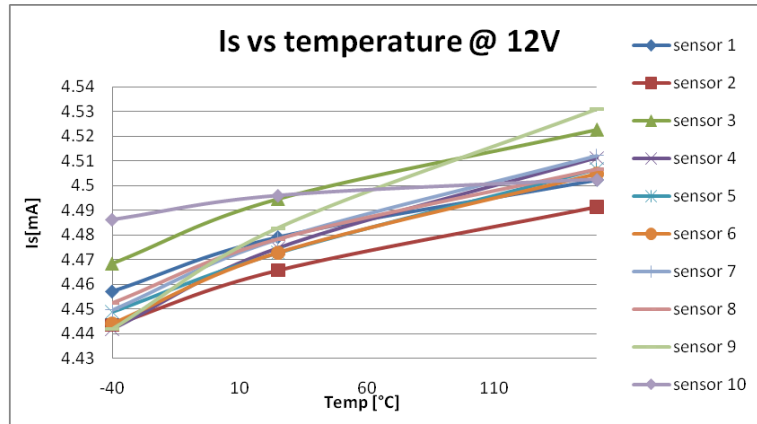


Fig. 8. Measurement results for 12V

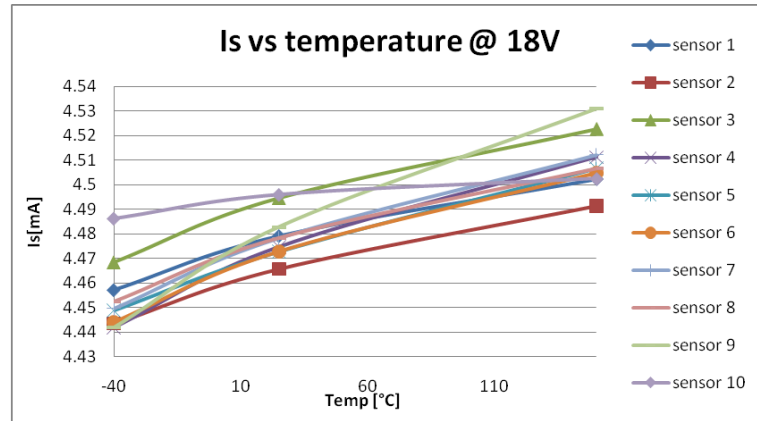


Fig. 9. Measurement results for 18V

The principle of the measurement is the following: first each desired temperature is set (in this case -40°C , 25°C and 150°C) and when it is reached and stable the supply voltage is varied for the desired values for the measurement (in this case 2,7V, 5V, 12V, 18V) and the current is measured [9].

As it can be observed, the experimental values are around the theoretical values but with a distribution. This distribution is caused by the technology, but also by the measurement error.

The purpose of the measurement on the magnetic sensor was mainly to confirm the reliability and functionality of the developed CEE. Comparing the manual measurements and the results from the simulation, the functionality of the developed environment was proven.

5. Conclusions

The automated setup implemented in this paper is easily configurable for many types of measurements and this was proven by the measurements performed on the two types of devices. The measurements on the magnetic sensor had the purpose of checking the functionality of the setup and the results are satisfactory. The measurements performed on the microcontroller were performed for various parameters like supply voltage and frequencies. Using these measurements we have created a simulator, which is providing an estimate of the power consumed by the microcontroller when executing an algorithm. The average error in the estimation of power consumption using the simulator is less than 1%, which demonstrates the accuracy of the measurements performed using CEE.

The magnetic sensor measurements show the reliability of the measurements and also the robustness of the CEE. The measurements were performed fast and with a minimum influence. The results show that the CEE offers a good method of characterization and the accuracy of the measurements is a good reason to be used in other applications also.

Acknowledgement

The work has been funded by the Sectoral Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labour, Family and Social Protection through the Financial Agreement POSDRU/6/1.5/S/19.

REFERENCES

- [1] *I.R. Walker*, "Reliability in Scientific Research: Improving the Dependability of Measurements, Calculations, Equipment and Software", Cambridge, United Kingdom, 2011
- [2] *Ronald Loui*, In praise of scripting, IEEE Computer, 2008
- [3] <http://www.microchip.com/>
- [4] <http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en538159>
- [5] *Mark Burns and Gordon W Roberts*, "An Introduction to Mixed-Signal IC Test and Measurement", Oxford University Press, USA, 2000
- [6] *V. Tiwari, S. Malik, A. Wolfe*, "Power analysis of embedded software: A first step towards software power minimization", Proceedings of the IEEE/ACM International Conference on Computer-aided design, pp. 384-390, USA, 1994
- [7] *R. Dochia, D. Bogdan, C. Burileanu*, "Model for software power estimation of an 8-bit microcontroller" Proceedings of the 34th International Semiconductor Conference, Romania, pages 443-447, 17-19th of October 2011
- [8] *Daniel P. Foty*, "MOSFET Modeling with SPICE - Principles and Practice", Prentice Hall, Upper Saddle River, NJ, 1997.
- [9] *J. Fraden*, "Handbook of modern sensors – Physics, Designs, and Applications", Springer, 2004