Open Source Oscilloscope for Hobby Users

Vlad Niculescu
Student
University Politehnica of Bucharest
Bucharest, Romania
vlad.niculescu@cetti.ro

Adrian Ioan Liţă
PhD Student
University Politehnica of Bucharest
Bucharest, Romania
adrian@axiplus.com

***Abstract* — The Internet of Things era is making hardware projects more accessible than ever to hobby and beginner users. As many of them work on their projects from home, there is a growing need for accessible tools, or even tools that they can build themselves for later user in other hardware projects.**

The main purpose of this paper is to present an open source solution for a portable oscilloscope, customized to meet the hobby working requirements. The small oscilloscope is very easy to use and communicates via USB port for processing, logging and display. This oscilloscope is very useful for any electronics applications, being a very handy tool for students and hobbyists.

The 2MSPS sampling speed and 8-bit resolution are considered to be enough for general-purpose hobby applications.

***Keywords* — oscilloscope, signal processing, DSP, C#, PIC, hobby.**

I. INTRODUCTION

The past decade has brought forward a revolution in hardware design, fast prototyping and DIY projects, backed-up both by the multitude of hardware and software solutions that make up the "Internet of Things" and by the multitude of educational materials, open source designs and courses available for students, young enthusiasts and hobby developers.

Open source hardware and software, such as the popular Arduino platform, or the open cloud platform proposed by Radovici et al. [1] have made hardware prototyping more accessible to the wide public. Educational resources include both new curricula developed in universities, on applied electronics and hardware concepts, such as the work proposed by Rosner et al. [2] or the experimental training proposed by Kane et al. in [3], as well as numerous open-source materials available un web sites dedicated for beginners and hobby developers.

There is however, a domain that the authors of the current paper consider that is not, yet, properly addressed: empowering hobby users to develop their own debug tools for their DIY projects.

Particularly, having an oscilloscope is a very useful thing for hardware prototyping and development, because it enables the user to debug, visualize and measure various signals. There are several solutions available; however, each one has specific limitations. Karim proposes a low-cost solution in [4] based on the Arduino platform, however, this is limited by the use of low resolution graphical liquid crystal display as final output. González et al. [5] propose an oscilloscope specially designed for educational purpose – but rather a complex implementation requiring advanced knowledge as well as close guidance for actual implementing the proposed design. Deng et al. propose a solution based on a sound card, with the advantage of a minimum of special hardware building required, but with a maximum frequency limited to 20KHz [6].

The present paper aims to offer a circuit that is easy to build, accurate, and affordable compared to other oscilloscopes available on the market. The performance of the proposed solution is also optimized considering the user's needs. A portable device may be, many times, a better choice than a laboratory oscilloscope, because you just plug it in the USB port, and it is ready to measure. The waveforms are captured easier on the computer, and new processing features can be added by customizing the PC application. In addition, many portable oscilloscopes have considerable measuring errors due to their cheap implementation of the input stages. To avoid this situation, only high quality components were included in the input circuitry.

II. THE HARDWARE BLOCKS

Fig.1 represents the block schematic of the oscilloscope, showing the signal route throughout the circuit. The microcontroller samples the signal received from input stage, and decides the signal's amplification or attenuation. The microcontroller is also responsible to implement the triggering mechanism. After the trigger detects the correct signal, sampling starts and a number of consecutive samples (limited by the memory) are stored. The samples are encapsulated in a 1600 elements packet, which is transmitted to the computer. After transmitting, the algorithm resets and the triggering/conversion process restarts. In order to avoid USB and PC latency problems, the transmission is not continuous, but requested by the PC every 300ms. A communication block is needed to translate the protocol of microcontroller (UART) into USB's protocol.

978-1-4673-8180-2/15/$31.00 ©2015 IEEE

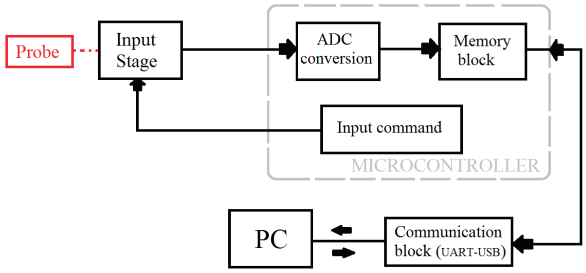

Fig.1 Block schematic

A. The input stage

The input stage process the original signal to meet all the input requirements of the ADC. If the signal's amplitude is too low, the waveform would appear in segments due to the lower resolution of the ADC. To prevent such situations, the signal is amplified. On the other hand, any signal above 5V (the maximum voltage allowed on the ADC pin) would damage the microcontroller, so an attenuation is required. The circuit in Fig.2 switches between four divider ratios: 10:1, 5:1, 2:1, 1:1. Integer numbers were preferred because is easier to make future computations. The MAX4052, an analog multiplexer with impressive performance from MAXIM, selects the enabled channel. Its input capacitance is only 2pF and the leakage current is 5nA. That is why the voltage ratio will not be affected, because the current through the resistors is significantly higher, and the low value parasite capacity will be compensated using the trimmers. The voltage dividers are parallelized with capacitors in order to obtain calibrated attenuators. The input impedance is always at least 1MOhm, which makes the oscilloscope highly competitive to the low-price USB scopes available on the market, such as DSO Nano V3: http://www.seeedstudio.com/wiki/DSO_Nano_v3.

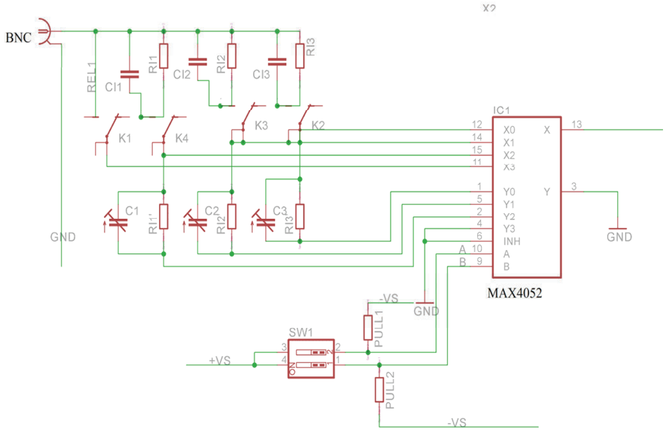

Fig.2. Divider switch

Fig.3 and Fig 4. presents the amplification stage. If low amplitude signals are measured, the first stage is set to 1:1, and the signal is going to be amplified. The OPA124 buffers MUX' output, and has an input current equal to 1pA.

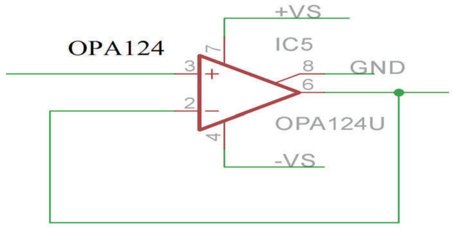

Fig.3 MUX buffer

The JFET input of the operational amplifier does not draw a current higher than 1pA for biasing, otherwise, the divider ratios would be modified and the signal altered. In order to set an amplification ratio, an analog multiplexer is connected on the feedback of the AD8055 to change the negative feedback value. The ratio is controlled by the MUXs, which is controlled by the microcontroller's digital outputs.

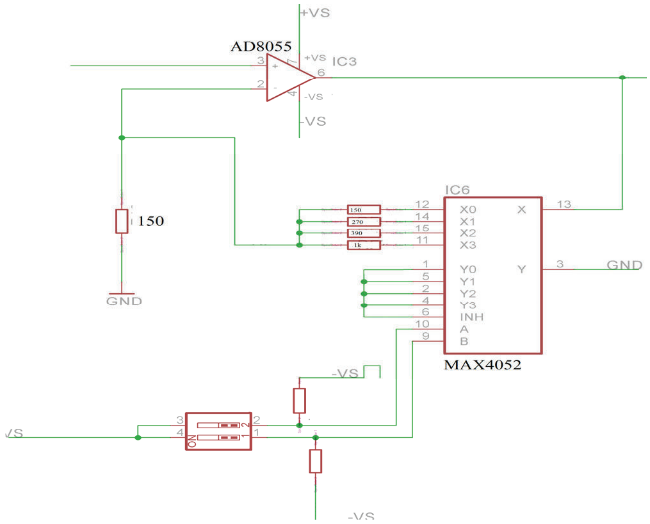

Fig.4. Vertical amplifier block

MAX477 is a high performance operational amplifier designed to work as a buffer, which feeds the ADC. It has a high input impedance (1MΩ) and a very low output impedance (<5Ω). The transistor is a level shifter to fit the signal between 0 and 3.3V.

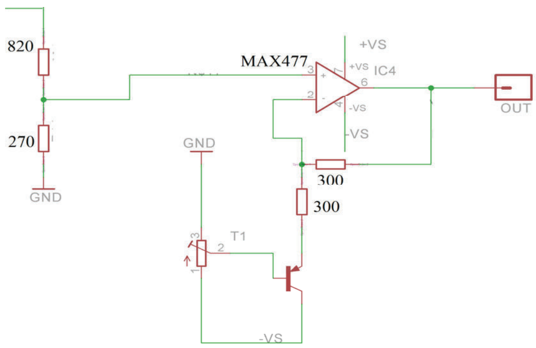

Fig.5. ADC buffer

B. The Step Down Regulator

The integrated circuit MCP16301 together with the rest of components represent a BUCK circuit, which efficiently lowers the the USB voltage to 3.3V in order to supply the microcontroller. High output capacitors attenuate the output noise.

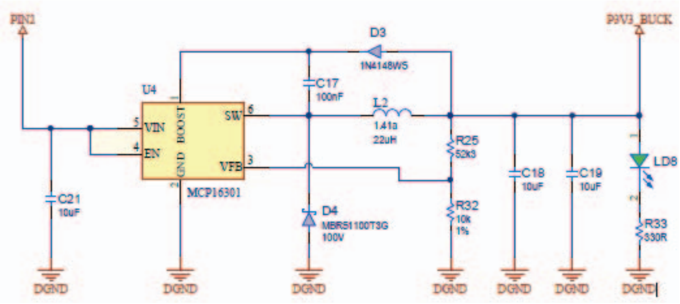

Fig. 6. Step down schematic

C. USB-UART Converter

The MCP2200 is the intermediary circuit between the computer's USB port and microcontroller's UART interface. This circuit was preferred because it can support a baud rate of 921600 with an error of just 0.16%.

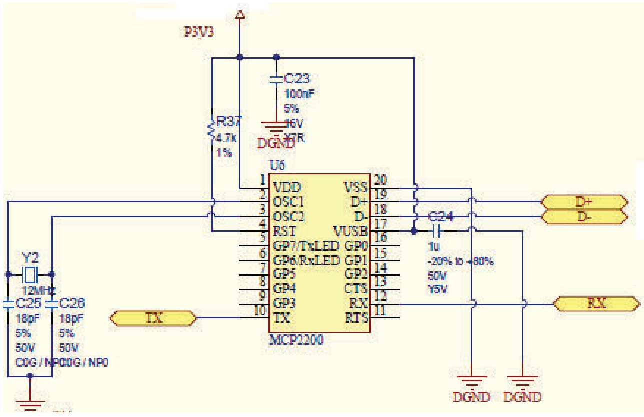

Fig. 7. UART-USB converter schematic

D. The ADC

As soon as the signal goes through the input stage, it arrives to the microcontroller's pin in order to be sampled. Because high and precise sampling rates are required, sampling cannot be done normally in the main program. The microcontroller has a smart feature for this action, which generates an internal high frequency square signal. Every positive edge of it generates a trigger which leads the program in the ADC interrupt. This assures the perfect timing and high frequency. In order to adjust the sampling rate the microcontroller adjusts the frequency of the square signal. When it comes to functionality, the A/D is SAR (successive approximation) based converter. This means that, in its structure, the converter has a DAC (digital to analog converter), a successive approximation register and a comparator. The precision of the approximation depends on the A/D converter's resolution, which is 10 bit for this PIC.

The microcontroller, dsPIC33FJ16GS504, was chosen especially for being equipped with this type of 2MSPS ADC, and also for its dedicated digital signal processing functions.

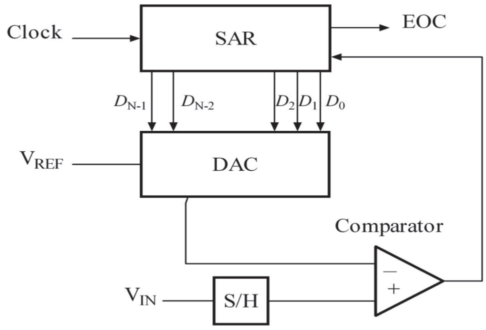

Fig. 8. ADC's architecture

III. THE SOFTWARE

A. The Sampling Algorithm

Because of the need to obtain high sampling rates, the samples cannot be transmitted after they have been read, because the transmission takes too long. Instead, a different algorithm is implemented: the microcontroller stores continuously samples, and when the memory is full it sends them as a packet. This resulted in a limitation commonly known: the limited amount of memory - just 2KB in the present case. In order to increase the amount of the samples, they were converted from 10 bits to 8 bits in order to occupy just 1B/sample. The controller has 2KB of RAM, and a few bytes are needed for other variables of the software. The final result is a packet of 1600 samples which are sent to the PC. In this way, a wide screenshot of the signal will be captured, even at the maximum sampling rate. The microcontroller's program is built on the model of a *state-machine*, respecting the basic principle that the system can be found just in one state at a certain moment on time. The states are: OFF_STATE, TON_STATE, ON_STATE, TOFF_STATE, WAIT_STATE. The first state initializes the software; TON_STATE enables the interrupts and prepares for going to ON_STATE, which is the main loop. Here the device checks if the 1600 points are sampled in order to create the packet. If a fault occurs, system goes in TOFF_STATE and WAIT_STATE, where the process is paused and faults are corrected before restarting.

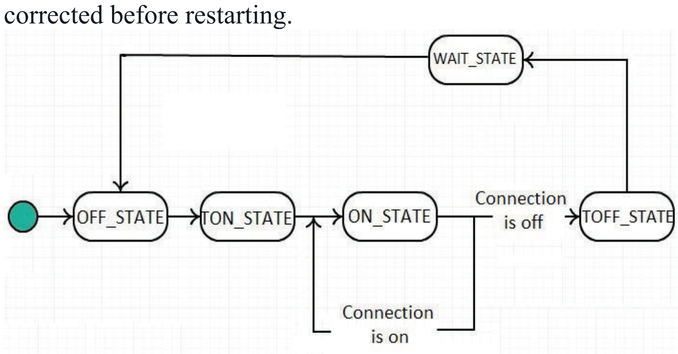

Fig. 9. The state machine

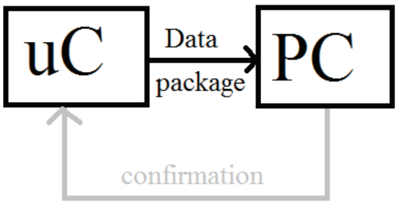

Fig.10. Commands route

In order to avoid communication congestions, the microcontroller was designed to start sampling only when the PC requests it to do so. This way it allows the PC to take care of the previous sample (process it and display on screen). Otherwise, the USB buffer would risk to overflow. The request issued by the PC also contains information regarding slope, trigger level and sampling speed, attenuation/amplification.

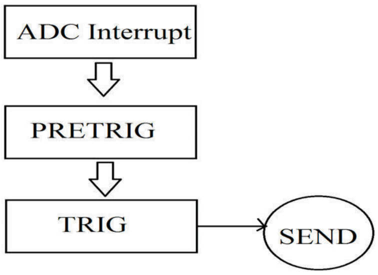

Fig.11. Data acquisition procedure

As soon as the packet is decoded, the loop opens and the square signal is enabled, with a period based on the sampling speed communicated by computer. The square signal generates an interrupt, which leads the software in PRETRIG state. Then, every sample value is compared with the trigger level until an equality is obtained. This signal's slope is also checked in this state, by interpreting the first derivative: positive or negative.

After the triggering conditions are met, the sampling and storing process is started until the 1600 length array is full.

B. Data packaging

As soon as the byte array is filled, the packing process is started. The packet has the format presented in Fig. 11. The format is chosen this way to ease the decoding process. The program was designed in such manner to always avoid blocking the state machine and its functionality.

Fig. 12. Package form

IV. EXPERIMENTAL RESULTS AND C# APPLICATION

A PC application was developed to communicate with the oscilloscope, setup the parameters and show the results. C# was the preferred programming language due to its highly available rich-UI elements.

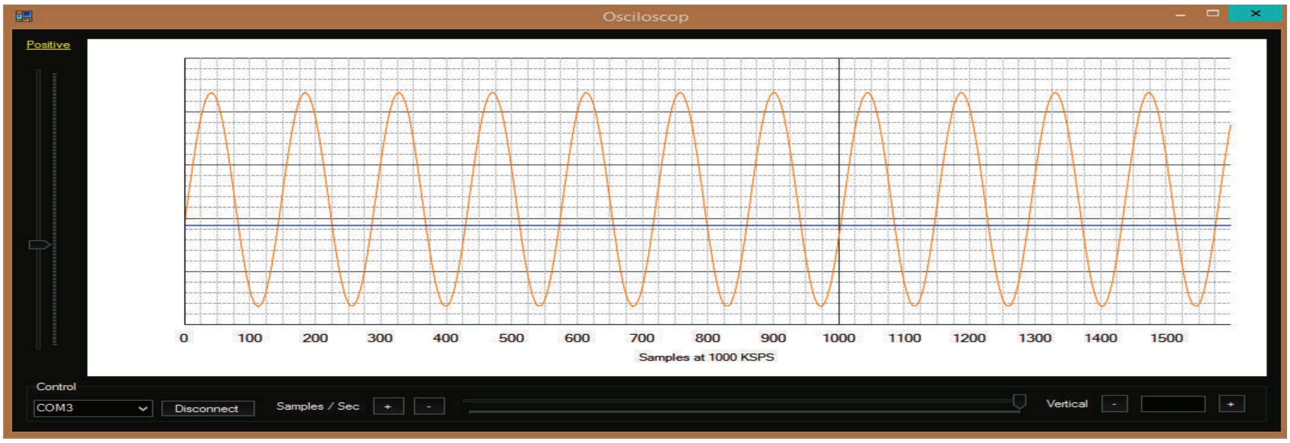

Fig. 13 Sinusoidal wave

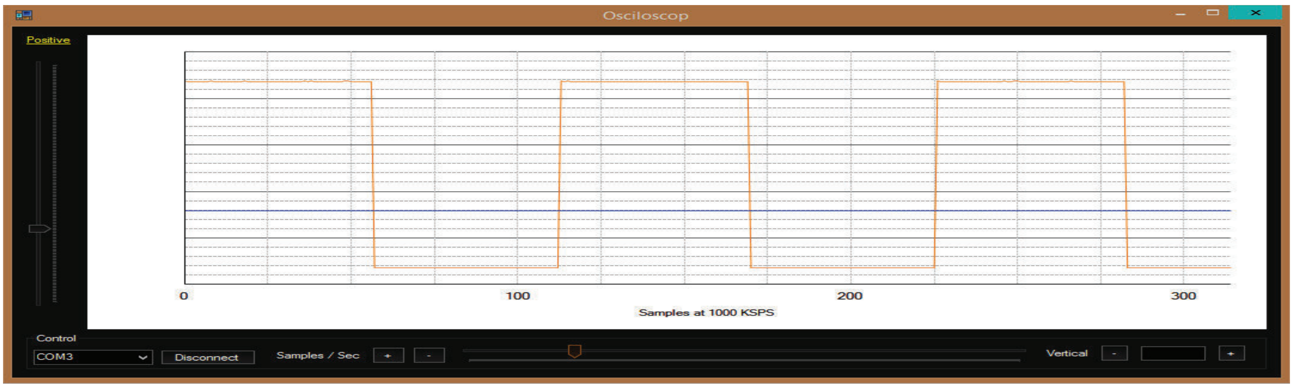

Fig.14 Rectangular wave

As soon as the application is started, the user inputs the baud rate and a COM in order to establish the connection to the oscilloscope. The interface provides adjustments for trigger level and time base modifications by using two track bars. A toggle button sets the slope. In order to support low and very low frequency signals, the sampling rate can be modified. The software uses three timers for managing the entire functionality: SerialTimer checks every 200ms if a new package is available and if so starts processing it. The second timer checks the communication and send a confirmation package to uC (with configuration parameters), giving it a flag to resend a new data pack. The third timer updates the interface when new data is available. The structure of the packet sent by the PC to the oscilloscope is:

snp[slope bit][trigger value][time base]end

Fig.15 Command package

The screenshot from Fig.13 was resulted by applying a sinusoidal wave of frequency 7Khz at the oscilloscope's input. The same measurement was done with a Tektronix TDS 2024, and comparing the parameters, it was obtained almost the same result for frequency and a maximum of five percent difference for amplitude. Fig.14 indicates the oscillogram of a rectangular wave, where the differences are identical to those in the previous case.

V. CONCLUSION

It is true that 2MSPS oscilloscopes are not very powerful oscilloscopes for industrial use, but for most hobby applications and for a lot of educational applications 2MSPS is deemed enough. Considering that the total cost of this device does not exceed $10, the proposed solution is also budget friendly for hobby users. Using a computer as display is definitely an advantage compared to a graphical LCD, because of its processing features. While a LCD is capable only to display the waveform, a C# application running under windows is able to compute advanced signal processing functions, like spectral analyses. The samples can be easily stored as well as imported in other applications for a future processing. For further development, the only limitations are the ADC's conversion rate and memory size. Replacing this by a higher speed ADCs and adding more memory can result in higher performance oscilloscope. Apart from other existing USB oscilloscopes, this is a very small one, being very easy to build due to its low number of components. Maybe, many hobby users prefer to give up a few percent of accuracy in order to get a pocket device more practical than a heavy laboratory oscilloscope. In order to eliminate the USB cable, the device can be equipped with a wireless module, using batteries for power supply. This would make the oscilloscope usable in the non-isolated high-voltage area. The presented method represents an optimum balance between complexity and performance, and it will successfully meet the requirements of a hobby user.

REFERENCES

[1] A. Radovici, I. Culic, "Open cloud platform for programming embedded systems", Networking in Education and Research, RoEduNet,vol. 12, pp. 1-2, 2013.

[2] Rosner, D.; Sarbu, G.; Tataroiu, R.; Deaconescu, R., "Applied electronics curriculum for Computer Science students," EUROCON - International Conference on Computer as a Tool (EUROCON), 2011 IEEE , vol., no., pp.1,4, 27-29 April 2011

[3] Kane, P.; Duda, M.; Farrell, S.; Jeffords, J.; Kimsey, T.; Rucinski, A.; Simon, M.; Jiawei Zhong, "Disruptive engineering training and education based on the Internet of Things," Teaching, Assessment and Learning for Engineering (TALE), 2013 IEEE International Conference on , vol., no., pp.632,636, 26-29 Aug. 2013

[4] Karim, I.A., "A low cost portable oscilloscope based on Arduino and GLCD," Informatics, Electronics & Vision (ICIEV), 2014 International Conference on , vol., no., pp.1,4, 23-24 May 2014

[5] González, J.J.; Odiwo, C.C., "Design of an Educational Oscilloscope: A hands-on learning tool," Education Engineering (EDUCON), 2010 IEEE, vol., no., pp.1873,1878, 14-16 April 2010

[6] Deng Yi, "Research and Design of Virtual Oscilloscope Based on Sound Card," Electrical and Control Engineering (ICECE), 2010 International Conference on , vol., no., pp.1566,1569, 25-27 June 2010

[7] P.R. Gray, P.J. Hurst, S.H. Lewis, R.G. Meyer, Analyses and Design of Analog Integrated Circuits, J.Wiley & Sons, 2001

[8] K.R. Laker, W.M.C. Sansen, Design of Analog Integrated Circuits and Systems, McGraw-Hill, New York, 1994

[9] Bob Orwiller, Oscilloscope Vertial Amplifiers, Beaverton, 1969

[10] Microchip dsPIC33FJ16GS504, datasheet and programming datasheet http://www.microchip.com/wwwproducts/Devices.aspx?product =dsPIC33FJ16GS504.

[11] T. Niculescu, "Study of transient phenomena using NI-USB data acquisition systems in Matlab Simulink medium on 64 bit operating systems", Journal of Advanced Computer Sciense & Technology, Article ID:4518, DOI: 10.14419/jacst.v4il.4518, pp.198-203, Impact Factor:0.9733

[12] D. Păsculescu, T. Niculescu, "Study of transient inductive-capacitive circuits using data acquisition systems", 15th SGEM GeoConference on Informatics, Geoinformatics and Remote Sensing, SGEM 2015 Conference Proceedings, ISBN 978-619-7105-34-6 / ISSN 1314-2704, Albena, Bulgaria, June 18-24, 2015, Vol. 1, pp. 323-329. DOI: 10.5593/SGEM2015/B21