

# 1. Introduction

---

## 1.1. Motivation

The present project was started in late 2012 as a support activity in training of the use of microprocessors in physics experiments, the topic given at the University of Ljubljana, Faculty of Mathematics and Physics. There are actually two mutually supportive topics on electronic acquisition of data and presented at the faculty. One is »Acquisition and processing of signals«, the other is »The use of microprocessors in physics experiments«. The first focuses on signal acquisition with PC compatible computer using a commercial acquisition system and basic signal processing algorithms, like the ones involved in sampling, filtering and evaluation of basic signal properties. The second transfers the algorithms to a new level by introducing hardware for signal acquisition, namely microprocessors and includes a touch of FPGA based systems, and additionally introduces more advanced techniques for data processing.

In order to support the lectured theory students are introduced into the laboratory where they use either PC-based signal acquisition systems (first course) or microprocessor based units (second course). Until 2012 the microprocessor used in these systems was Texas Instruments' MSP430F1611, the 16bit unit with built-in memory, ADC, DAC, two counter/timer units ... A board using the MSP430F1611 was developed in 2007, and used since. The board's details can be found elsewhere on these pages (only in Slovenian language). However, the limitations of this processor became clear after the introduction of more sophisticated and complex algorithms in the course. The MSP430 series is optimized for low power consumption, and as such does not support either large data arrays or high speed processing. It has been demonstrated that a simple FIR filtering is limited to sampling frequencies below 1 kHz due to slow multiplication. Additionally, and to a regret of the designer, the lack of proper ADC input driving on the board resulted in poor resolution of the ADC, being only about 9 bits. Combining the properties of the processor used with the flops in the design limited the usefulness of the board to a demonstration level. The board was used for teaching but often with comments like "this is supposed to work as ...".

The proper solution to the problem is to introduce a new microprocessor based board of a better design. The newly designed board should at least have all the units included in the former design, but should use a contemporary processor, and should be good enough to be used in some serious work in laboratories at the faculty not just for demonstration purposes, but also for signal processing and computer interfacing. The board should be simple (EURO size, double layer at most, simple connectors for input & output signals, simple power supply, provisions to box the board into a standard enclosure, provisions to connect the board to PC computer using the USB cable, provisions to connect various sensors to the board, ...).

The standard (and cheaper) solution would be to look on the market for a DEMO board which includes all units and ports required. There are several boards implementing several microcomputers available. However, the author of the first design (MSP430) is a hardware addict, and enjoys designing and building electronic circuits. There was no way to (self) convince the author into not building its own board. The board and its use will be described in the articles that will follow. Hopefully this will encourage other enthusiasts to re-build and contribute, as will the students to experiment and learn. The board may even find a way into other laboratories and experiments.

## 1.2. The requirements for the selected microprocessor

First of all the board should be based on a contemporary (fast) microcontroller, preferably 32 bits wide. The microcontroller should house enough RAM to hold the digitized signals (more than 10 kB as available in the MSP430F1611), and should have decent amount of ROM for program. The ROM should be “flash” type allowing it to be infinitely reprogrammable (well, not really infinite times but as experiments within a classroom are concerned infinite is a nice number to start with). There are several such microcontrollers available and are produced by many companies.

The board should be equipped with analog input lines to measure delicate analog signals with reasonable resolution. An ADC with 12 bits of resolution, supplemented with a DAC with the same resolution is nice to start with. The combination of these two allows the on-line digital processing of analog signals. The ADC can be used to sample the analog signal, and pass digital values to microprocessor for calculations. The results of calculations are outputted using the DAC back into analog world. At least two analog signals will have to be sampled, preferably more. This requires an analog multiplexer in front of the ADC, and at least two DACs since multiplexing at DAC output is not an option. The ADC/DAC requirement already reduces the number of processors to select from.

Additionally, the ADC and DAC should be fast to allow digitalization of at least stereo audio signals; this is expected to boost motivation for the use. Therefore a sub 10  $\mu$ s conversion time is a necessity. Many experiments in physics are based on exact phase relationships between two or more signals. The phase relation calculations require simultaneous sampling at two or more channels and limit the usefulness of multiplexer in front of ADC. The microprocessor selected should have at least two separate ADCs to allow simultaneous sampling at two channels, and should have at least two DAC with option for simultaneous updating as well. This requirement significantly reduces the number of processors to select from.

The sampling of the ADC should be periodical, the period being defined by a quartz crystal oscillator and the divider built inside the microcontroller. Most microprocessors are equipped with such module called a timer. The timer should be able either to initiate conversion at the ADC or to trigger an interrupt to the microprocessor in many different ways, so a complex timer is preferred. If possible there should be more than one timer available. Timers are used also for counting of pulses, measuring time marks and generating digital signals. For instance a pulse width modulation (PWM) is popular for controlling the power delivered to a load. Many microcontrollers offer such timers, but less microcontrollers offer several such timers.

The ADC and DAC require periodic attention of the processor to transfer the data to and from these units. With high sampling rate the processor spends most of the transferring data from ADC and to DAC instead of calculating. A special hardware can be implemented into the microcontroller to ease the loading of the processor. A devoted memory can be used to form a circular or

intermediate buffer for data from ADC or to DAC. Alternatively, a process called Direct Memory Access (DMA) can be implemented in hardware of the microcontroller to release the processor of the data-moving task. The selected microcontroller should implement either of the two, and this also limits the number of microcontrollers available.

Sensors used in contemporary control circuit are often already equipped with ADCs and have digital outputs in form of few electrical lines called a bus. The communication over a bus conforms to a predefined set of rules described in a protocol. Protocols like I2C, SPI, RS232, USB and alike are commonly used, and the microcontroller selected should implement as many as possible of such protocols so that the programmer will not have to code the communication protocol into the software but will instead use the built-in hardware modules to implement the communication protocol without the interaction of the software. Additionally, the microcontroller should have ample number of pins devoted to the connection of sensors. This requirement also significantly reduces the number of microcontroller suitable for implementation.

Last but not least the programming of the microcontroller should be simple (low cost), since it will be used in school. The programmer should not cost more than, say, 30€. The software for writing the program should be free at least in the extent to allow the writing of school programs that are usually some kB in length. The software should have a “professional” look and feel, and should allow the compilation, linking and transfer of the program into microcontroller, as well as the debugging, single stepping, the use of breakpoints, memory and variable inspection, register peeking, manual control of microcontroller registers and output lines, and more.

### **1.3. The requirements for the board**

The board should be EURO size and should fit into a standard enclosure. The board itself should be simple two layer design, and preferably one layer should be used as a ground plane to improve analog performance. The connections for microprocessor signals should be located at the edges of the board to allow easy access for experimenting. It is expected that the board as such will be fully exposed to students during the experimenting, and that the board anyway cannot be fully protected. After all, this is experimenting: if it is fool-proof, than it is not experimenting at all (please feel free to disagree with the above), so only a basic protection circuitry will be implemented. This opinion is based also on the past experience with the MSP430 based board, where in five years of experimenting only one microcontroller died due to a tweezers falling onto a board...

The connectors at the board should be cheap and easily repairable. Pin stripes as used for IC sockets can be used, since they are easily replaceable, and wires of correct diameter can be pushed into holes and stay there. When all wires with un-appropriate diameter are removed from a laboratory, the use of such sockets is safe. Instead of the IC pin stripes regular headers can be soldered into the board, providing a better connection for instance to the connectors on the front plate of the optional housing. The jumpers on the board should also be cheap and replaceable, so IC pin stripes or regular headers can be used.

The analog input lines can have fixed sensitivity, but must be able to allow the measurement of input signals of either polarity. The input can be made selectable by the means of a jumper: single polarity between 0 and +3 V and bipolar -3 V to +3 V. Analog output lines should be single or dual polarity as well, the selection can be done by jumpers.

The board should be equipped with some push-buttons to allow the interaction of the experimenter with the software. The board should also be equipped with a small LCD display to allow the presentation of results to the experimenter. Both pushbuttons and the display should be connected in a way that allows the connection of separate external components when the board is mounted into an enclosure.

The power supply of the board should be a universal one. A non-regulated power supply can be used, and the board should have local voltage regulator to supply the microcontroller. The local voltage regulator should be protected against reversed polarity of the power connectors. Alternatively, the power could be supplied from a USB connector plugged into a PC computer.

The board should be equipped with some type of power output. The output should be sufficient to allow the connection of at least two small DC motors or two stepper motors simultaneously. The same power output lines could be used to supply power to other loads using the PWM technique. There should be some auxiliary power amplifiers on the board, like power MOSFETs. The maximum current for each power output line should be in the range of 0.5A at least, and the transistors should be configured as open-drain circuits.

#### **1.4. The final decision on the microcontroller and board design**

Having the above requirement in mind a board was designed and built. It is based on ST microcontroller STM32F407 in 100pin LQFP package. The package was selected for simple soldering. The STM32F407 is a 32 bit ARM CORTEX-M4 architecture microcontroller with 512 kB flash memory and 192 kB of RAM. It runs with internal clock of up-to 168 MHz, and has lots of goodies built in, among them all the required ones. The details of the microcontroller are available at: <http://www.st.com/internet/mcu/product/252149.jsp>. Yes, the author is aware that there are numerous microcontrollers available that fulfill the requirements. However, the author opted for this particular chip. The decision was not based on former experience with this processor or this family of processors. The decision was not based on options not stated in the requirements. The decision was mostly based on a review of specifications of the available components and the support provided by companies, as well as the price for the finalized board. If the author would spend more time deciding then the decision might have been a different one, and the author is not willing to spend time arguing he made an optimal decision.

The basis for the project was the STM32F4DISCOVERY board, sold by ST for a small price of about 14€ including the programmer (yes, this board also eased the decision on the chip). The initial idea was to use this board and make it sit in a “piggyback” (whatever this means literally) position on a homemade board with the required connectors and power drivers. The DISCOVERY board, regrettably, proved to offer poor analog performance due to its poor layout. The DISCOVERY board can still be used as a programmer since it is cheap and well supported. It can also be used where analog performance is not critical. However, it is without the protective casing, and one might prefer to use regular programmer like ST-link/v2, priced at about 25€ (<http://www.st.com/internet/evalboard/product/251168.jsp>).

The STM32F407 uses 3.3 V power supply at a current depending on the peripherals used and the clock speed. When only the CPU is running at full speed of 168 MHz, the current consumption is about 50 mA; with all peripherals enabled the current consumption increases to about 100 mA. Details are given at the chip’s datasheet.

The STM32F4xx series has 5 V tolerant inputs. This means that regardless of the power supply a signal between 0 V and 5 V can be connected to any input of a microcontroller and no harm will be done. If one exaggerates and connects a pin to a signal outside of these limits, then the consequences will have to be corrected. The chip seems to be considered protected against reasonable amounts of static electricity, since also the STM32F4DISCOVERY board does not implement any protection against the static electricity, and there are no noticeable warnings to the potential user. However, regular protection against static electricity is welcomed, and should be followed. The details on “electrical hardness” can be found in the STM datasheet.

The allowed regular output current on all digital outputs from STM32F4xx series is limited to few mA as in any digital circuit. In order to protect the chip against the over-current when input voltage exceeds the normal 0 V to 5 V range a series resistor is inserted in all digital lines that are available to the user. The prototype uses resistors of either 100  $\Omega$  or 33  $\Omega$ , therefore an over-voltage of 0.5V (for 100 $\Omega$  resistor in series) will do no harm (guaranteed, the datasheet allows 5 mA of “injected” current). Where this is considered not sufficient, a higher value resistor could be inserted bearing in mind that such resistor not only limits the regular output current but also the speed of transition from one logical level to the other and may influence the behavior of the circuit.

The analog input circuitry will be based on the use of operational amplifier in buffer configuration. Fast, rail-to-rail IO chips will be used. The power supply for these chips will be made selectable by the means of jumpers. One may use either 0 V / +3.3 V power to guarantee the safety of the microcontroller input pins. However, such power supply of the operational amplifiers reduces the range of input voltages to the ADC within the microcontroller. Some tens of mV are stripped at either supply line, limiting the ADC range from about 50 mV to 3.25 V. Where this cannot be tolerated, the power supply for the buffer chips can be increased to  $\pm 5$  V assuring the full 0 V to 3.3 V at the input to the ADC. However, with this the microcontroller can be harmed by the overvoltage at microcontroller pins, so a double diode should be used to divert the over-current to either power supply. The additional  $\pm 5$  V power supply can be provided by a local DC-DC switching regulator.

The analog input lines should be selectable to allow either the measurement of positive signals or bipolar signals. A simple voltage divider will be placed in front of the buffer amplifier to allow the selection. The analog output lines will be buffered using a operational amplifier. The same as stated for the analog input buffer regarding the power supply and limits also applies here.

The digital output lines will be as many as applicable, and will be available as chunks of microcontroller ports. The chunks will be ordered sequentially and will be as big as possible. The connectors where these chunks are available will be expanded by several ground and power pins to allow the connection and power supply of additional circuit. All digital output lines will be protected by a series resistor of the selected value, see above. The same lines can also be used as inputs.

The LCD used will be a standard 16 characters by 2 lines display. It will be mounted onto the board using long spacers to allow components to be placed underneath. The LCD will be connected to the board using a header; the data transfer to the LCD will use a standard bus but will only be 4 bits wide. The placement of the LCD should not reduce the accessibility of any vital component.

There will be six pushbuttons on the board. They are provided with pull-up / pull-down resistors to ease the use for those not aware of pull-ups / pull-downs built into the microcontroller. Additionally, there will be a provision to connect external keyboard using a standard header

There will be some LEDs on the board. These will be connected to spare outputs of the microcontroller to allow the signaling of vital microcontroller states as defined in the user software.

There will be two driver chips for power output signals. The selected units are L293D, which host two full H-bridge configurations. They can provide up-to 1A output current, and can be supplied with maximum of 25V. The chips have protective diodes built-in, and can be paralleled in the case of emergency. The control signals for the drivers are taken from the PWM outputs of timers built inside the microcontrollers, but can be controlled by the software as well. The same control signals are also available at a connector to be used by an alternative power driver if needed.

There will be a mix of connectors to support various communication protocols: I2C, SPI, RS232 (TTL and  $\pm 12V$  versions). There will be a provision for USB connection to the personal computer. The last can be optically isolated to reduce the interference from the computer.

There will be 4 MOSFET power transistors on the board, each with separate input and output. The transistors are BSP296 type, allowing the switching of up-to 1A at up-to 500V (not both limits simultaneously, please; see transistor specifications).

This sums up the requirements and the decisions made, and now it is time to see the board.