# Experimenting with

## Low-cost development system with a USB interface

Emile van de Logt, MA MSc (The Netherlands)

**Texas Instruments supplies handy USB evaluation sticks with related software for its low-cost MSP430 controllers. Unfortunately the I/O facilities are somewhat limited. These can be substantially enhanced with the help of the experimenter's board described here. This combination forms the hardware basis for a mini-course 'Starting with embedded C', which can be found elsewhere in this issue.**

Sometimes several initiatives converge at just the right time to create a new concept. For some time Rotterdam University had been looking for a low-cost development system for its students in Automotive and Electronic Engineering, which could be put to use in microcontroller tuition. In addition, for logistical reasons Elektor was looking for a more practical replacement for the very popular E-blocks for its Embedded C Programming workshops. So both Rotterdam University and Elektor were effectively looking for the same thing, although for different reasons.
Once contact was established between Rotterdam University and the Elektor labs, it didn't take long before it was decided to combine forces. The advantages of both the E-blocks as well as the configuration used by Rotterdam University were examined and a joint specification was produced.

For students it is obviously important to obtain the hardware and software as cheaply as possible. The accompanying development system should ideally be included free of charge. It should also be easy to use so that new students can quickly create their first program. Preferably not in assembly language, but in C using a full-featured C compiler! It should be enjoyable to work with, so preferably it should be able to create sound (buzzer), display numbers (7-segment display), flash (LEDs), work via the USB port rather than the parallel port (modern laptops no longer have these) and also include further expansion possibilities (I$^2$C, SPI). It would also be nice not to use an 8-bit system any more, but rather something with a bit more muscle (16 or 32 bits).

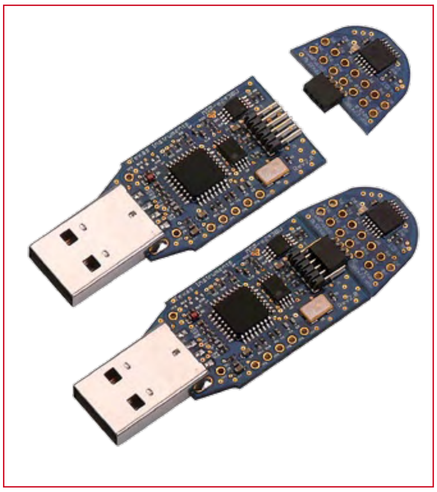


Figure 1. The eZ430 evaluation stick, around which this experimenter's system is based.

### History

Not long ago an average electronics department would have to raise an expense request to obtain such a microcontroller system. These systems were often large dedicated computers, which required expensive software suites to develop embedded applications. Via an ICE (In-Circuit Emulator) the program could then be debugged on the target system.
If we said at the time that we could obtain such a system for less than 50 euros (or the equivalent in pounds or dollars) we would have been met with looks of disbelief. Despite this, there are currently students walking around with such a system in their backpack and they have lessons in programming such a microcontroller!

### The MSP430 series

All the big electronics manufacturers supply microcontrollers offering a wide range of functions. However,

# the MSP430

```
** Author  : Bert Korthof
** Date    : 26-2-2009
```

```
    P1OUT = 28;  // all pins high
    for (i = 0; i < 10000; i++) ; //delay
    P1OUT = 16;  // all pins high
    for (i = 0; i < 10000; i++) ; //delay
    P1OUT = 8;  // all pins high
    for (i = 0; i < 10000; i++) ; //delay
    P1OUT = 24;  // all pins low
    for (j = 0; j < 10000; j++) ; //delay
  } // while()
} // main
```

when the above criteria are taken into account there is one manufacturer that stands out a bit from the rest, and that is Texas Instruments (TI). The MSP430 series in particular consists of a range of full-featured microcontrollers with a large number of I/O facilities. The most important properties are that they require very little power to run and that they contain a 16-bit processor core.

To keep things as simple as possible for the average electronic engineer, TI has designed evaluation 'boards' for this type of controller, where the complete hardware environment is housed inside a USB stick (**Figure 1**). This hardware environment is known as the eZ430. To this you can connect (via a so-called Spy-Bi-Wire connection) various target boards.

A complete system like this, including a C compiler, can be obtained for the unbelievably low price of under £20! After installing the software and plugging in the USB stick you can immediately start with programming this fascinating microcontroller.

But are there any disadvantages with this system? Unfortunately there are, since the eZ430 system only has limited I/O facilities. The target board inside the USB stick is very small and therefore has only a single LED and one connector. This is not enough if it is to be used as a training tool or in microcontroller workshops.

## Specifications

- Experimenter's board with several I/O possibilities
- Powerful 16-bit MSP430F2012 controller running at 16 MHz, 2 KB Flash and 128 bytes RAM
- 4 indicator LEDs
- 7-segment display
- Piezo buzzer
- Three pushbuttons
- I²C/SPI connector
- Powered via the USB stick or an external adapter

## The Elektor MSP430 board

There are basically two ways in which the number of I/O facilities can be increased in this system:

1. Add a new I/O board to the target board using the existing connector.

2. Design a new target board with the number of required I/O facilities and connect this to the USB stick using the Spy-Bi-Wire interface.

For various reasons it was decided to go with the second option. A separate I/O board using a connector isn't very stable and could sometimes give rise to connection problems. Furthermore, the target board is cheap enough that it makes little difference to the total cost whether the microcontroller is or isn't part of the I/O board.

The circuit diagram of the MSP430 board designed by Elektor and the lecturers from Rotterdam University can be seen in **Figure 2**. The most important part on this I/O board is of course the microcontroller itself, an MSP430-F2012 (IC2). The reason for using this particular microcontroller and not the F2013 as supplied by TI in the eZ430-F2013 kit is that the F2012 has a different type of A/D converter that has a greater range. Apart from that, both microcontrollers are identical.

The experimenter's board has a set of three pushbuttons that are connected

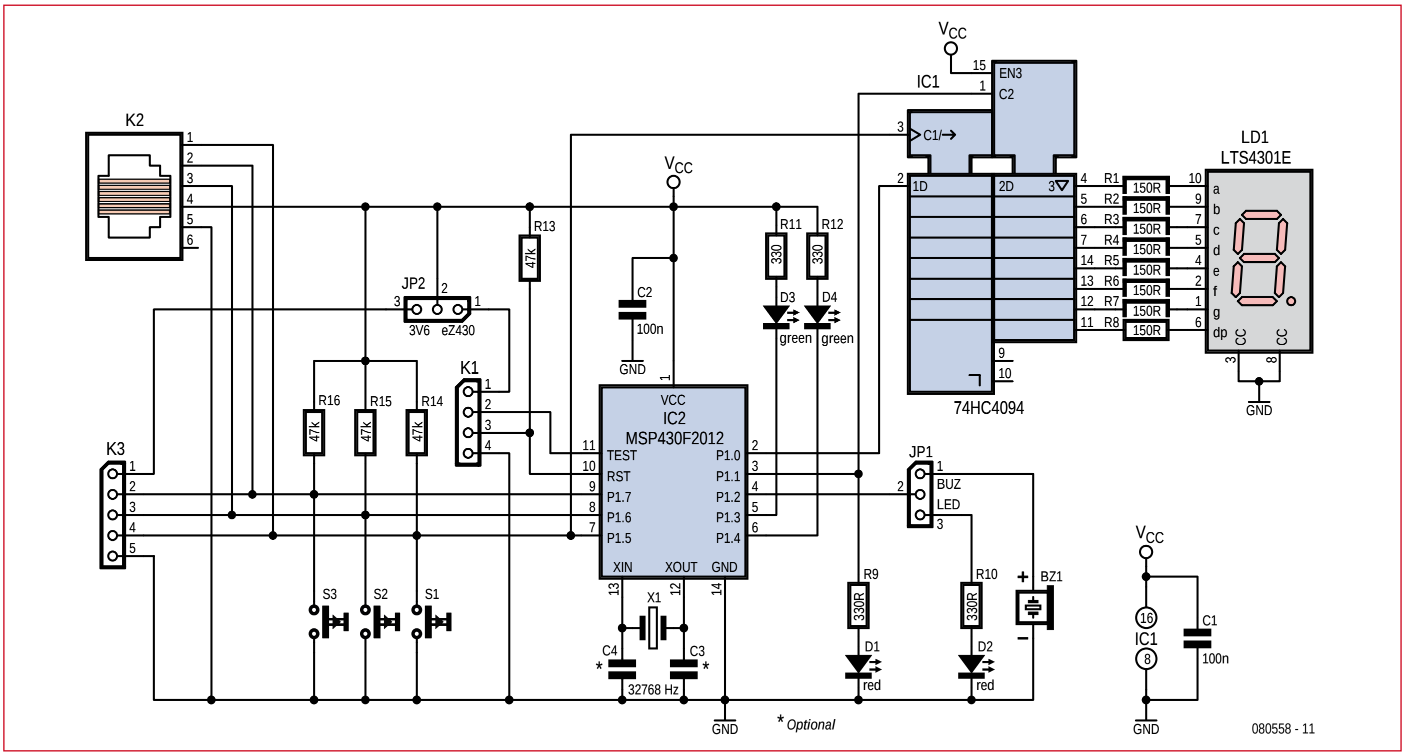


Figure 2. The circuit diagram for the MSP430 experimenter's board.

to P1.5, P1.6 and P1.7. Most of the pins actually have several functions, depending on how they've been configured via the software. Four LEDs have also been included, two green and two red. The green LEDs are connected to the positive supply via a resistor and the red LEDs are connected to GND via a resistor. To light up a green a logical zero has to be programmed on the relevant I/O pin. For red LEDs this is the exact opposite: with these a logical one has to be programmed. It was done this way to quickly give students an insight into the differences of the common configurations that can be found.

JP1 can be used to manually select either the red LED (D2) or the buzzer (BZ1). When the buzzer has been selected via JP1, it is possible to create a tone by quickly switching between logic zero and logic one on pin P1.2. Increasing or decreasing the rate at which the level switches causes the frequency of the tone to change.

7-segment display LD1 is connected to the microcontroller via shift register IC1 (a 74HC4094). To display a digit on LD1 you first need to send an 8-bit code serially to IC1, after which input C2 of IC1 is made high via P1.1. The serial transmission of data to IC1 is done via pins P1.5 (clock) and P1.0 (data), where the most significant bit is sent first.

The board gets its 3.3 V supply voltage via the Spy-Bi-Wire interface (K3). There is therefore no need for a separate power supply, except in cases where the board is used in stand-alone applications. In those cases the setting of JP2 has to be changed so that the 3.3 V supply voltage can come via K3 (pins 1 and 5).

We have also thought about expansion possibilities. The microcontroller has a Universal Serial Interface (USI), which can function as either an SPI or an I²C serial communications interface. These signals have been made available on both K2 (standard I²C connector) as well as K3 (expansion connector). The required I/O pins are listed in **Table 1**. We haven't described the A/D converter yet. The F2012 has a built-in 10-bit analogue-to-digital converter with eight input channels. It is also able to read the state of an internal temperature sensor and measure the value of the supply voltage.

**Table 1.**
**I/O pins for external communications**

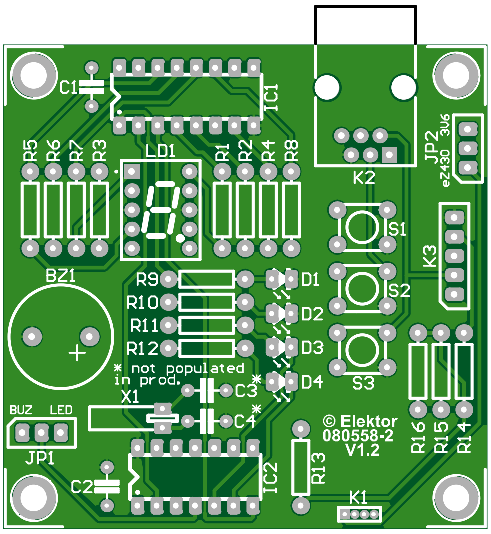
| I/O pin | SPI | I²C | ADC |
|---|---|---|---|
| P1.5 | clock | | ADC 5 |
| P1.6 | serial data out | clock | ADC 6 |
| P1.7 | serial data in | serial data | ADC 7 |

## Construction

Now that we've given an overview of the experimenter's board and its various functions, it's time to put this I/O board to work. We decided early on to populate the I/O with standard components and not SMDs, so that the construction wouldn't present any problems for less experienced constructors.

In **Figure 3** you can see the PCB that was designed for the I/O board. Soldering all the parts should be quite straightforward. As usual you should take care with the polarity of the LEDs and ICs. Elektor also supplies a completely populated board, including the sometimes difficult to obtain connector for the connection with the USB stick (Elektor Shop # 080558-91). The EZ430 USB stick can also be obtained from Elektor (Elektor Shop # 080558-92).

The board is connected to the USB



## COMPONENT LIST

### Resistors

R1-R8 = 150Ω 0.25W
R9-R12 = 330Ω 0.25W
R13-R16 = 47kΩ 0.25W

### Capacitors

C1,C2 = 100nF
C3,C4 = optional, not fitted here (see TI Appl. Report SLAA322)

### Semiconductors

D1,D2 = low-current LED, red, diam. 3mm
D3,D4 = low-current LED, green, diam. 3mm
LD1 = 7-segment LED display, common cathode (e.g. Lite-On LTS4301E)
IC1 = 74HC4094
IC2 = MSP430F2012IN (TI)

### Miscellaneous

S1,S2,S3 = PCB mount pushbutton, 5x5mm (e.g. Tyco FSM4JH)
K1 = right angled 4-pin connector, lead pitch 1.27mm (Mill-Max # 851-93-004-20-001000)
K2 = 6-way RJ11 connector, PCB mount (Molex # 95009-2661)
K3 = 5-way SIL pinheader
JP1,JP2 = 3-way pinheader and jumper
BZ1 = passive piezo buzzer (e.g. Kingstate# KPEG242)
X1 = 32.768kHz quartz crystal
16-pin IC socket for IC1
14-pin IC socket for IC2
PCB, # 080558-2
Ready assembled and tested board: Elektor Shop # 080558-91
TI eZ430-F2013 evaluation kit: Elektor Shop # 080558-92

Figure 3. PCB layout for the board. Mini-connector K1 is used to connect the USB stick (via the middle four pins!).

stick via the Spy-Bi-Wire interface. It's easiest if you open the plastic housing of the stick and remove the circuit boards; the target board can then be removed and in its place you can connect the experimenter's board.

## In practice

Does it work? Certainly! Although in this context by 'work' we don't mean if the hardware functions correctly, since that won't be a problem in most cases. Instead we mean if it works well in an educational environment, where students have to familiarise themselves with the complexities of programming in the C language on embedded systems.

The results, both at the Rotterdam University as well as in the Embedded C Programming workshops at Elektor, are very positive. It appears that participants can quickly learn to write simple programs in the C language and can get them to work on the experimenter's board. Most of the participants of these courses won't have had much experience in programming in C, but it isn't long before they start to delve into complex tasks such as writing timer interrupts, creating functions and driving the hardware on board. In subsequent projects, for courses both in Automotive as well as Electronic Engineering, we often come across the same hardware again!

In the Embedded C programming workshop of Elektor a completed and tested version of the board is used, obviously in conjunction with the eZ430 kit from TI. This complete kit can also be ordered by students, so there is no need to order them separately.

## Do-it-yourself

After reading this article we wouldn't be surprised if many readers would also like to try their hand at embedded C with the help of this experimenter's board.

A description of the software needed to drive all hardware in this I/O board will be covered in a short three-part course, again in collaboration with Rotterdam University. The first article can be found in this issue of Elektor. It covers the first few steps: the installation of the development environment and the testing of the completed board using the first example program.

(080558-I)

## The author

Emile van de Logt is an Electronic Engineering training manager at Rotterdam University.

He studied Electronic Engineering at the Technical University in Eindhoven and Management Studies at the Open University.

Emile spends his spare time designing electronic circuits and he is an amateur beer brewer. He also takes care of the Embedded C Programming workshop and the FPGA-VHDL workshop for Elektor.