The Next Generation of Microcontroller Development Tools

Try the ghostly approach

By **Liam Walton** (UK)

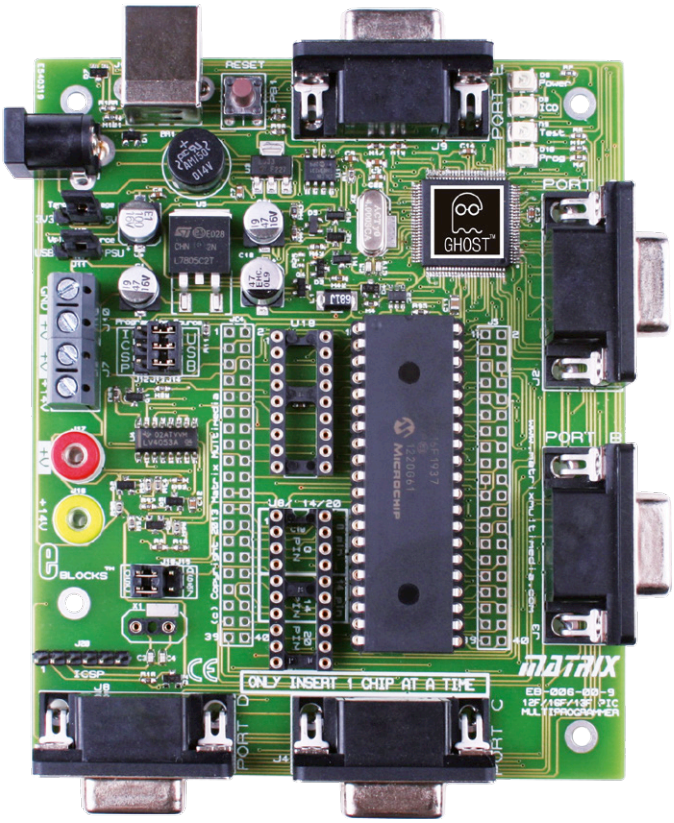

Figure 1. The new PICmicro multiprogrammer with Ghost functionality.

Microcontroller compilers, programmers and devices are now commodity items. In this article brought to you by Elektor.POST we look at one company's efforts to leapfrog the competition and set the standards for the next generation of development tools: 'Ghost' technology.

One of the great things about being involved in technology is that it is always changing. All of our readers will be very aware of the huge changes in the microcontroller development tool market over the last five years as the number of microcontrollers used in electronics has expanded. Well hold on to you hats because there is a new cycle of technology that is going to change everything again.

Enter a Ghost

Regular Elektor readers will be familiar with Flowcode and E-blocks which have been available on the Elektor store for a few years now. Now from the same company that develops this technology—Matrix Multimedia in the UK—comes something new: 'Ghost' (**Figure 1**). The idea behind Ghost is simple: Ghost watches you whilst you develop your projects. To do

this the developers have created an operating system on a powerful 16-bit microcontroller with an external block of RAM that sits on the programmer board. The microcontroller has two functions: first it programs a target chip with a hex file under the control of a host PC. Second, it provides utilities that monitor the microcontroller whilst the program is executed. Some of these utilities are fairly straightforward: multi-channel logic analyzer, multi-channel oscilloscope etc. Other utilities provided are perhaps unfamiliar to all other than professional coders and we will get to those later.

So how does this work?

Well it's simple really. The Ghost controller is based on a 100-pin quad flat pack chip with more input output lines than you can shake a stick at. The target microcontroller has a maximum of 35 input output lines. Every pin of the target microcontroller is connected to a single pin of the host microcontroller via a high value resistor which ensures that target circuitry is not affected by host circuitry. Whilst the program runs on the target the host records the changes in the status of the target pins and logs them in the on-board 4-megabit static RAM device. Ghost sends this data to the PC on request for further processing. Simple. The Ghost operating system also provides other functions like pausing the target chip during program execution, dumping register values, verifying chip contents, and other functions that make the developer's life easier. Of course once the data is back in the PC it needs to be processed and presented.

At the PC side

The PC side of Ghost resides within Flowcode which has recently been updated to version 6

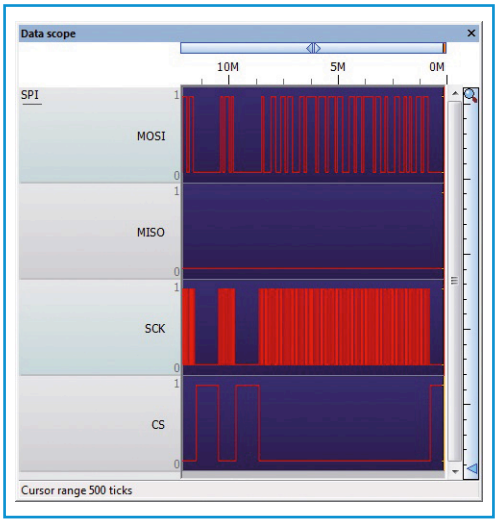

Figure 2. Ghost target chip pin status data displayed on the Softscope.

[1]. This new version of Flowcode includes not only a range of chip compilers for popular devices like PIC, ARM and Arduino, but in addition a fully specified interpreter language: a complete suite of software that runs on a Windows PC. This is all based on flowcharts with the goal being to make program generation as easy as possible.

So once our Ghost data is in the PC users have a choice: either to use some of the standard utilities provided, or to write brand new utilities which manipulate test and debug data in whichever way suits them. First let's take a look at some of the standard ones.

Ghostly utility 1: Oscilloscope

The tool of choice for many engineers is the oscilloscope. So the first tool of interest to us is the Softscope. You can see this in **Figure 2**. The Softscope can show multiple analog and digital traces at the same time. An example of its analog capabilities is shown in **Figure 3**.

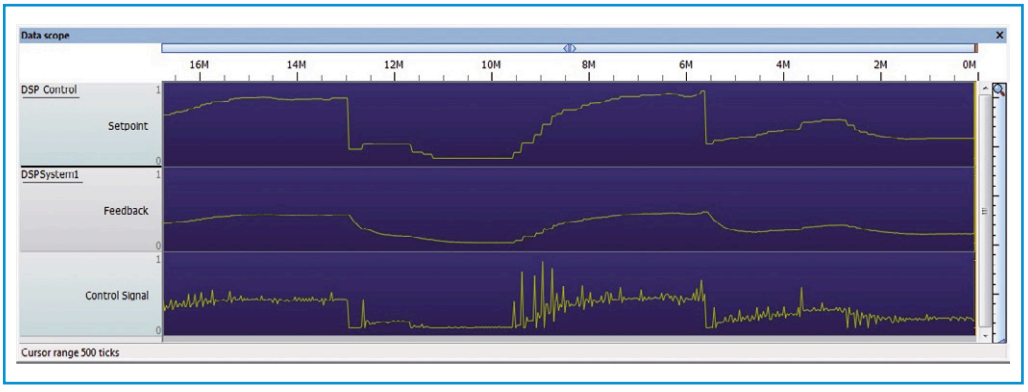

Figure 3.
Analog Ghost data on the Softscope.

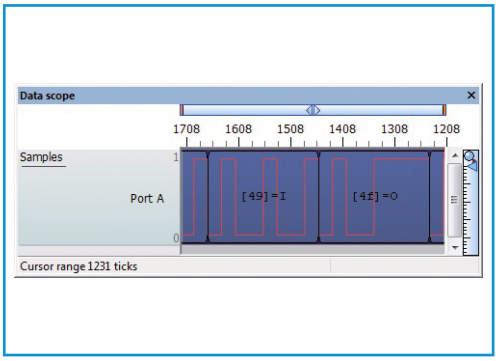

Figure 4. An overlay to the data showing its ASCII representation.

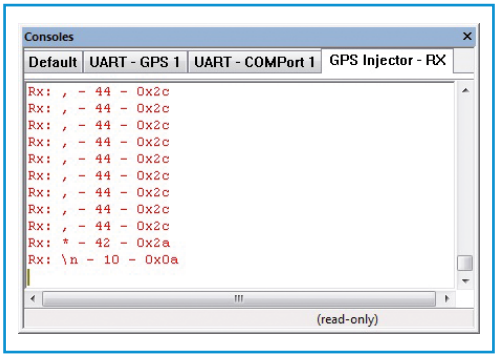

Figure 5. The Console displays low-level Ghost data in a scrolling text format.

You can effectively connect as many traces as you like to the Softscope using a properties window. Currently Ghost is not selective—it records all data on all pins and you can review it at your leisure and at different 'time bases' using the sliders above and to the right of the Softscope. The Softscope also allows you to add a communications system overlay to the traces so that you can see the ASCII equivalent communications data on the pins you have selected (**Figure 4**). Currently this is limited to USART protocols like RS232, SPI, I²C but the developers hope to expand this soon.

Ghostly utility 2: Console

The next utility Ghost, provides is a 'Console' which allows you to see interpret target pin data as sequential messages in ASCII format (**Figure 5**). This is a fantastic utility

if you are developing anything involving communications. Now Ghost starts to get really clever: The PC side interpreter decodes these messages in a separate Console window and displays them as system specific strings: for example, in **Figure 6** you can see the ASCII decoded to show GPS string data. At the same time you have complete PC side control of other windows and objects in the system so that you can see every level of data coming in, you can verify that you are decoding data correctly and sending out data correctly.
Flowcode provides further debug and test features in the form of off the shelf and custom components that use Ghost data in a human friendly format after it has been processed by PC-side programs. In **Figure 7** you can see that GPS data can be transformed to show speed on a dial, location on a world map and on the hardware LC display.

Conclusion

The next generation of microcontroller development tools will include a host of technologies beyond simple compilers and programmers. The combination of hardware based data gathering and programming circuitry combined with user programmable PC side interpreters will provide a range of tools to allow engineers to go further, faster, and will allow those with relatively little experience of coding to develop highly functional electronic systems.

(130582)

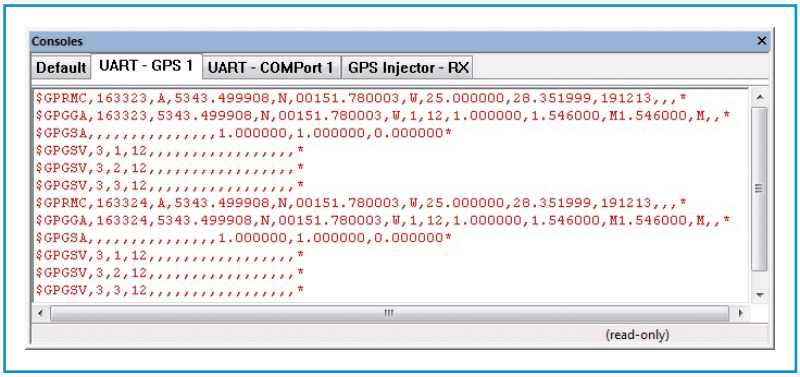

Figure 6.
Decoded GPS string data displayed on the Console

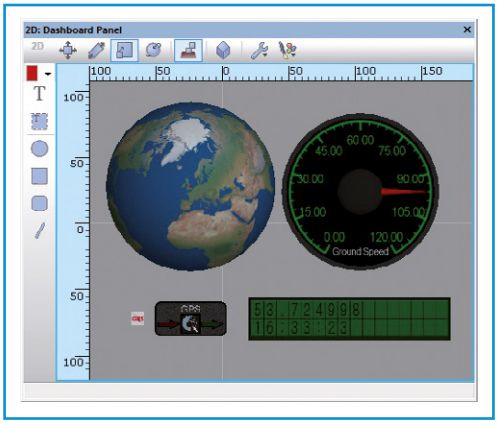

Figure 7.
Ghost data is transformed into a debugging Human Machine Interface using Flowcode interpreter commands.

Web Link

[1] www.elektor.com/development/eblocks-flowcode