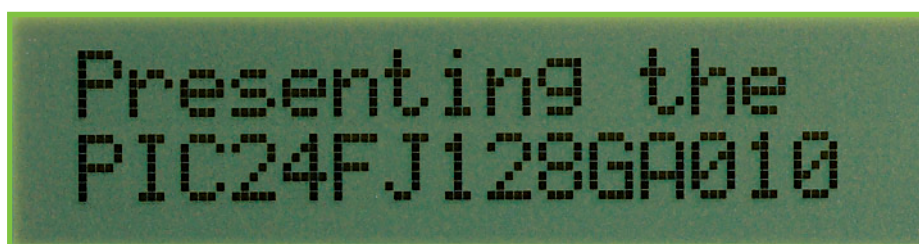


Explorer-16 (3)

Part 3: Speaking Thermometer, CF card simulation and Crypto Puzzle

Jan Buiting & Luc Lemmens, in cooperation with Microchip Technology and Labcenter Electronics



More advanced (but still 100% free) simulation this month

with a VSM model for media storage card devices added to our PIC24F system (all virtual, of course). To cap it all, solve a CF card crypto puzzle and win fantastic prizes sponsored by Microchip and Labcenter. But let's first revert to last month's Speaking Thermometer project.

Last month's Speaking Thermometer is not just a simulation you can run using three main components: (1) Labcenter's Proteus VSM, (2) Microchip Technology's MPLAB (both from your Explorer-16 CD-ROM) and (3) Elektor's archive file called *Demo2.zip*. All components, by the way, are free, the CD-ROM having come as a gift with the full print run of our January 2007 issue.

If you haven't done so already, please do run the January 2007 'Demo1' simulation of the Explorer-16 board itself. In the virtual environment created by Proteus VSM, press buttons and click away on other controls to your heart's content and see how the system responds with its various indicators.

'Demo2' from the February 2007 issue is funnier to see... and hear in actual use! Particularly if you simulate temperature variations and manage to create your own voice files. The aim of these simulations is to

teach you how both simple and complex PIC24F MCU software can be created and debugged to the highest possible level in combination with hardware devices like sensors and voice output (how's that for an actuator device). The hardware/software interaction can be put through its paces in VSM/MPLAB without having real components cluttering your desk or even appearing on the boss' Project Purchasing sheets. If

the simulations teach you how both simple and complex PIC24F MCU software can be created and debugged

you need still stronger encouragement: with simulation your PC does most of the design work for you, saving time and money compared to working with hardware components to achieve a design goal.

While awe stricken with the power of the simulations and the degree of integration between the Labcenter and Microchip Technology products, let's not

forget that the Speaking Thermometer design can be made to work in hardware too. For this, all you need to know is how to program the PIC24F PIM on the Explorer-16 demo board that's part of our **Explorer-16 Value Pack**.

Speaking Thermometer

The following is described to help you become acquainted with the hardware contained in the Explorer-16 Value Pack. Extensive product descriptions, user manuals and software installation guides being supplied on CD-ROMs in the Pack (take your time to read it all, it's worth the effort), we can concentrate on our specific requirements. An up to speed description is given assuming of course that you are reasonably conversant with PCs.

1. Plug the Audio PICtail board into the PICtail bus connector on the Explorer-16 board. Check that the PIC24F PIM is fitted on the demo board.

2. Plug the PICkit 2 module onto the 6-way pinheader on the Explorer-16 board **observing that the LEDs are at the side of the Audio PICtail Plus board** (see note supplied with Explorer-16 Value Pack).

3. Apply power to the Explorer-16 board (9-15 VDC at about 300 mA).

4. Install the software that comes with the PICkit 2 from your Explorer-16 Value Pack. Check that version 2.0 or higher is available as earlier versions do not have support for the PIC24F MCU series. If necessary, updates can be downloaded from the Explorer-16 project page on the Elektor website.

5. Plug the USB lead from PICkit 2 into a free USB outlet on your PC. Install the device by responding to the usual prompts and windows that appear. PICkit 2 will be automatically recognised from then on.

6. Run the PICkit 2 software (a shortcut will be available on your desktop). The program will report: 'PICkit2 found', and, importantly, 'PIC Device found' (in this case, the PIC24FJ128GA010 on the demo board).

7. Now click File → Import hex → Demo2.hex. You may need to Browse your system if you can't remember where you put the .hex file (see part 2).

Once the file has been found and selected, the PICkit 2 program says 'Hex file successfully imported' and you'll see the usual controls for a programmer like Write, Verify, Read, Erase. There is no need to activate either of the 'VDD Target' boxes.

8. Now click on Write and the hex file is sent off to the PIC24F target device on the demo board (**Figure 1**). The red LED in the PICkit 2 module will flash to indicate the system is busy. Wait for the process to complete.

9. Remove power from the demo board, wait a few seconds and power on again. The Speaking Thermometer will start and you will hear the "degrees Celsius" message from the mini loudspeaker on the Audio PICtail board. In some cases, the reset button has to be pressed to launch the application.

10. The current temperature can be requested by pressing pushbutton S4 on the demo board.

Explorer-16 project highlights

- Economic and educational gateway to 16-bit microcontroller technology and simulation
- Explorer-16 Value Pack supplied at unbeatable price
- Free Proteus VSM, MPLAB IDE and MPLAB C30 supplied on CD-ROM
- Free simulation project files with instalments
- Explorer-16 demo board accommodates PIC24F/H and dsPIC33
- PICkit 2 suitable for PIC baseline / midrange / 18F / 24F / dsPIC33 (updates available)
- PIC24F and dsPIC33 PIM modules available for standalone applications
- Written and supported by experts at Microchip Technology, Labcenter and Elektor
- Crypto Competition
- Interactive support through project page and forum on Elektor

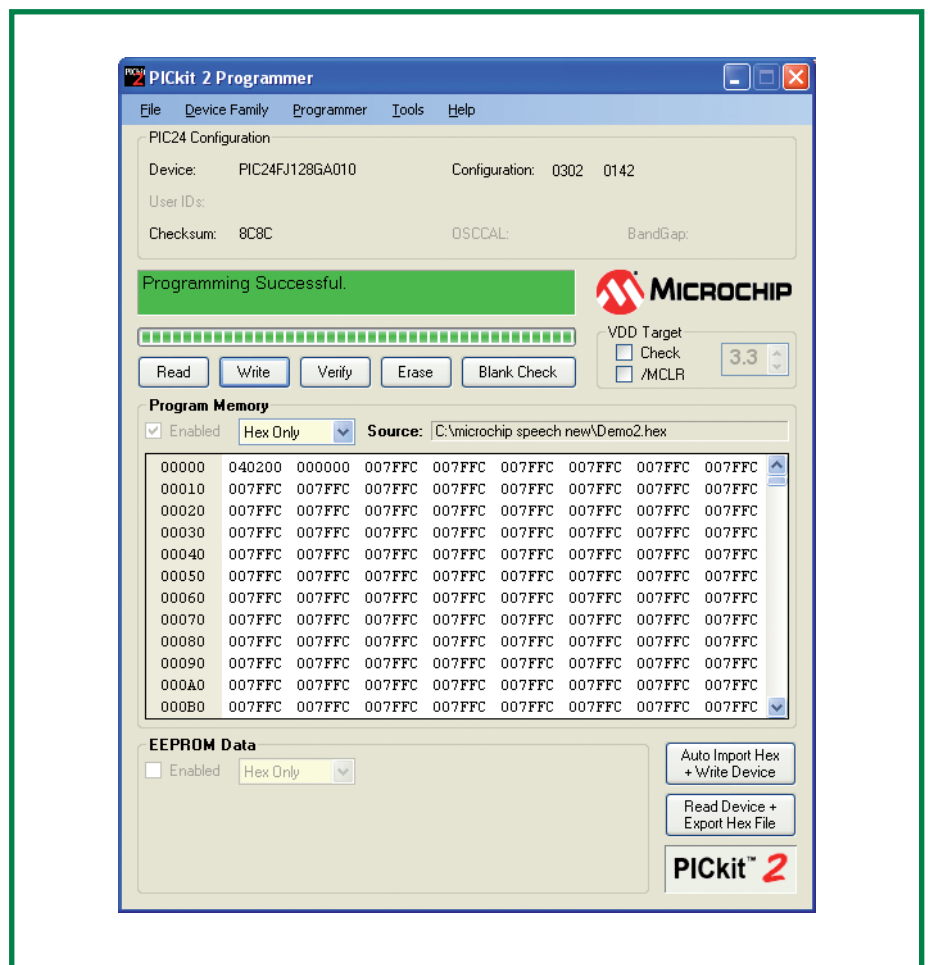


Figure 1. PICkit 2 has successfully loaded the hex object code file for the Speaking Thermometer and is ready to transfer it to the PIC24F target device via the plug-on pod.

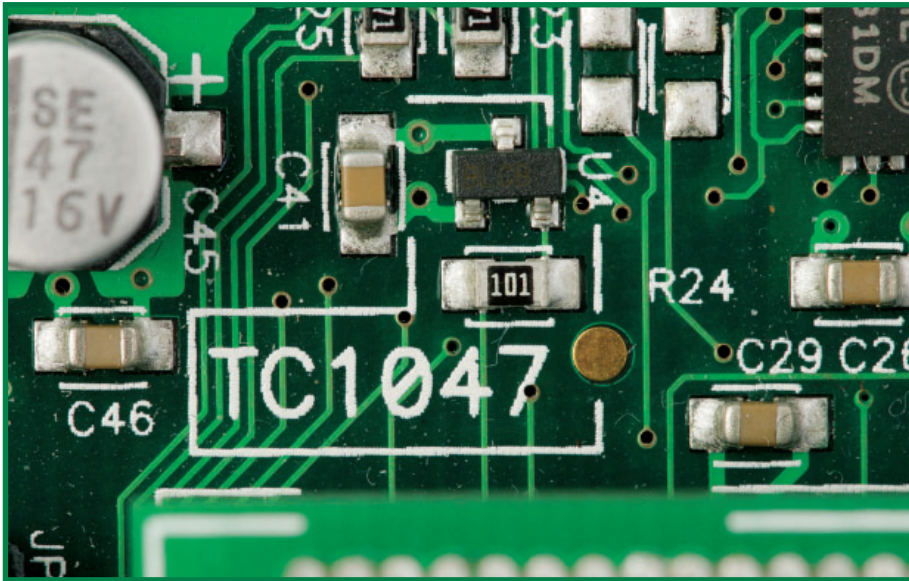


Figure 2. TC1047 temperature sensor on the Explorer-16 demo board. It is fully supported by the system software and easily addressed in C.

For better sound reproduction, connect a set of powered PC loudspeakers to the jack on the Audio PICtail board.

TC1047

So where is the temperature sensor? Although not shown on last month's arrowed floorplan of the Explorer-16 demo board, it is actually near the PIM, see **Figure 2**. The tiny TC1047 device proves the simplicity of connecting sensors to a PIC24F MCU. The TC1047 datasheet is worth examining for your own projects hence it is available from the Explorer-16 project page.

Put your fingertip on the three-pin SMD for about five seconds and press S4 again.

An educational gadget

The Speaking Thermometer may look trivial but has a lot of potential for some really advanced programming in C. A code snippet of the program is shown in **Figure 3**. How about these challenges:

1. change the program to start speaking automatically when a temperature change of more than half a degree C is detected. Ditto, make an alarm sound when a certain temperature is reached.
2. create your own voice files. A tutorial on how to do this will be available from the Free Downloads Area on the Explorer-16 project page. You will find that this requires extra storage memory and the section below on CF cards will prove useful.

```

PORTAbits.RA3 = 0;           // Enable audio amplifier

if(PORTDbits.RD13)         // If pushbutton has been released
{
    while(!AD1CON1bits.DONE); // Perform A/D conversion on TC1047A
    Temperature = ((ADC1BUF0*10)/31)-50; // Calculate temperature

    // Convert temperature value into the speech files to play
    if(Temperature/100)     // If hundreds non-zero
    {
        filename[0] = '1'; // filename = "100.dat"
        filename[1] = '0';
        filename[2] = '0';
        filename[3] = '.';
        filename[4] = 'd';
        filename[5] = 'a';
        filename[6] = 't';
        filename[7] = 0;
        hFile = MPFSOpen(filename); // Open file
        PlayClip(hFile);           // Play audio clip
        Temperature -= 100;        // Decrement by 100 leaves only tens/ones digits
        for(time=0;time<65534;time++); // Delay
    }

    if(Temperature>19)       // If remaining two digits are > 19

```

Figure 3. Extract of the C source code written for the Speaking Thermometer. Here, the ADC is read and the temperature translated to speech.

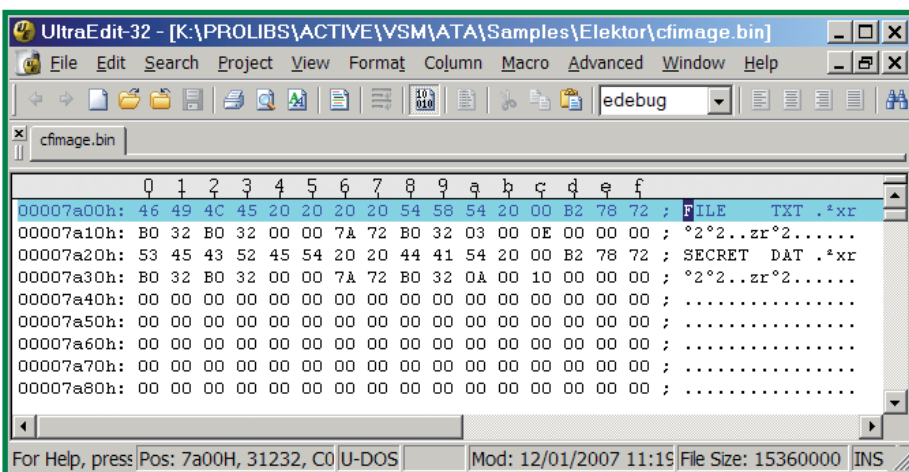


Figure 4. Directory table on a (virtual) Compact Flash card. The location 0x7A00 applies just to this sample image produced using UltraEdit.

As you rise to these challenges you will be making extensive use the C30 compiler, Proteus VSM and the MPLAB environment. Remember, **everything can be simulated**, boosting confidence that the real thing will work spot-on. There is a lot to experiment with — let us know how you get on and post your findings in our online Forum.

Onwards with media storage cards

Now, let's move on to working with media storage cards using the PIC24F MCU and the Microchip Media Storage Library.

The virtual hardware for this part of the article is supplied and designed to be used with the demonstration ver-

CSI Miami needs help COMPACT FLASH CRYPTO COMPETITION

The compact Flash image (cfimage.bin) used in this month's Proteus VSM simulation was actually taken from a real CF card that belonged to a secret agent. It contains file a called 'SECRET.DAT' which in turn contains contact details leading to a limited source of free Microchip ICD2 debuggers and Proteus VSM software.

The first three readers of each of the four main language editions of Elektor (12 prizes in total) able to extract the information contained in the file and follow the instructions therein will be able to obtain a free ICD2 debugger and a copy of Proteus VSM for the PIC24FJ128GA010.*

Of course, the file 'SECRET.DAT' has been scrambled, but the secret agent was careless and left another file on the CF card that may prove helpful in cracking the code. All the information you need to perform a simple forensic examination of the card image has been supplied in this article, and you should not need to use any specialist knowledge of cryptography.

Via the Message Area on the Explorer-16 project page we may or may not post some clues to cracking the code and getting at the information leading to the prizes. Have fun!

Competition not open to employees of Microchip Technology, Labcenter Electronics and Segment b.v., its business partners and/or associated publishing houses.

sion of Proteus VSM supplied on the Explorer-16 CD that came with your January 2007 issue.

The design consists of a PIC24F, a simulated compact Flash card and an RS232 terminal. This is broadly equivalent to plugging in the Compact Flash PICtail plus card into the Explorer-16 demo board, and then attaching an RS232 terminal or terminal emulator to UART2 – in real life.

Also supplied is an example project, DEMO3.MCW, which makes use of the Microchip Media Storage Library. This library provides a simple file I/O API and implements an FAT16 storage structure on the media card. The Media Storage Library can work with both Compact Flash and SD/MMC cards, although in this article we shall be using it with Compact Flash only. Useful background information on

Compact Flash (CF) cards and the FAT16 format may be found in References [1], [2] and [3]. Things you need to know about these media include

- the fact that a CF card is run in ATA mode whereas an SD card is run in MMC mode;
- the Sector organisation of an ATA drive;

the secret agent was careless and left another file on the CF card that may prove helpful in cracking the code

- the need for a filing system;
- the basics of FAT16 – directory tables etc.;
- the file CFIMAGE.BIN is a binary dump of the drive sectors.

Figure 4 shows a hex dump of the directory table at 7A00h. A lot of clues

on how the system is implemented in software can be obtained from it. Note that the exact location of the directory table will vary according to the size of a CF card.

Talking to the card in C

Provided you are reasonably conversant with the C programming language, you can review the file FileIO.c contained in Demo3.zip for each of the functions listed above. This will tell you how the various

technical requirements for CF cards could be implemented for our PIC24F system. It should be noted that the Media Storage Library is still in the development phase at Microchip and the relevant disclaimers should be observed. Although Labcenter and Microchip have worked closely together to be able to of-

