

# Flowcode And The Microchip PICkit 2

By John Dobson



## Using Flowcode to programme PICs with the PICkit 2

**M**ICROCHIP'S PICkit 2 Debug Express with the 44-pin Demo Board, reviewed in the June 2007 edition of *EPE*, must be in line for a prize as the world's smallest PIC programmer.

Equally impressive is Matrix Multimedia's Flowcode programming language, which allows you to dispense with MPLAB IDE and provides you with an intuitive flowchart programming interface.

This article offers an introduction to programming PIC microcontrollers using the free version of Flowcode (limited to 2K of compiled code), and shows you how to use it to create a simple LED flashing program for the PICkit 2 44-pin Demo Board. This should allow you to get to grips with the basic operation of Flowcode and PICkit 2.

### Flowchart

Program design often starts with a flowchart, planning the sequence of steps which the microprocessor will follow. With Flowcode, it is a very simple matter to convert the flowchart into a Flowcode program.

This first example develops a system which outputs one of two lighting patterns, depending on whether or not a switch is pressed. You can see this in Fig.1 (opposite).

Let's now convert this into a Flowcode program.

A free demo version of Flowcode is available from: <http://www.matrixmultimedia.com/temp/FlowcodeV3.exe>. The complete software is available from *EPE*. It is also available as part of a *Special Package Offer* bundled with TINA Pro V7 (Basic), the circuit simulation, testing and PCB design software – see the *CD-ROMs For Electronics* pages in this issue.

When you install Flowcode, make sure that the 'Microchip PICkit 2' option is selected.

- Run the Flowcode application.
- Click on the 'OK' button when the reminder screen opens.
- Next, you have the option of creating a new Flowcode flowchart, or opening an existing one. Select the option to create a new one.

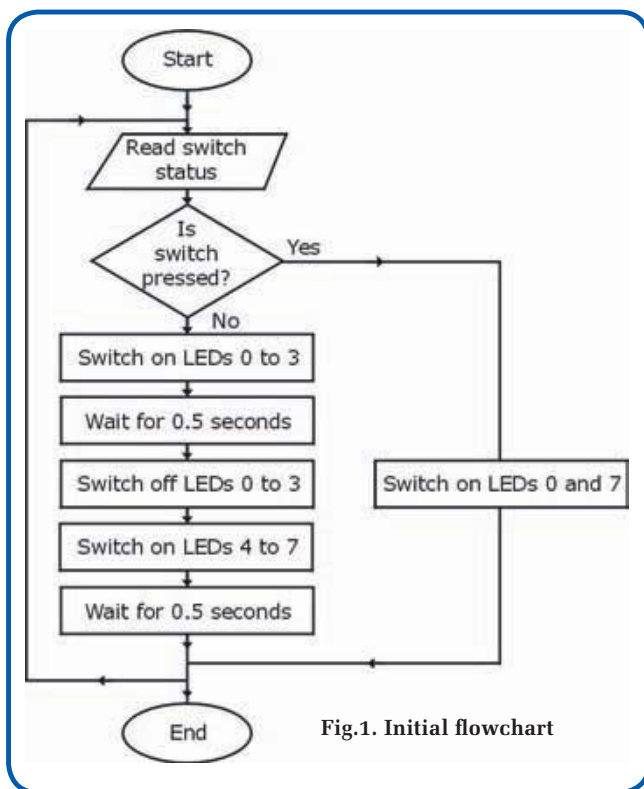


Fig.1. Initial flowchart

- You now have to choose a target PIC microcontroller. The 44-pin Demo Board is fitted with a PIC16F887, so choose that from the list.
- A new workspace called 'Main' opens.
- Click and drag icons from the Icons Toolbar running down the left-hand edge of the screen, to make the Flowcode flowchart shown in Fig.2. For the moment, do not worry about configuring them – we will do that next.

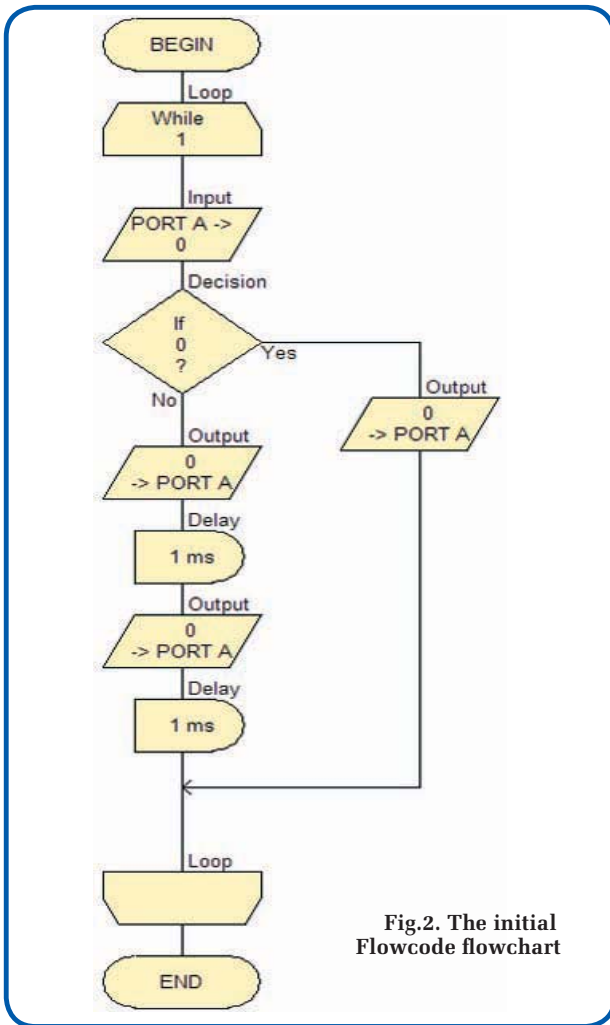
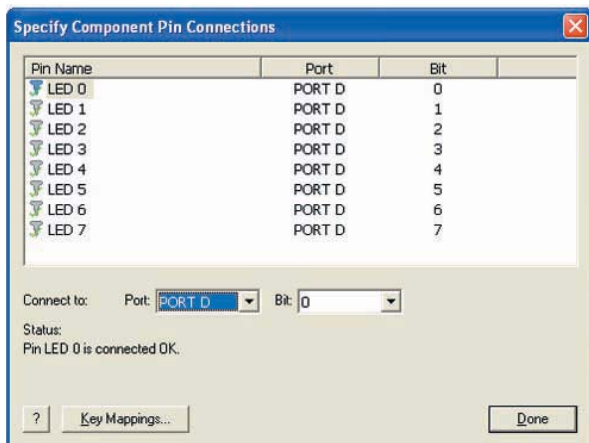


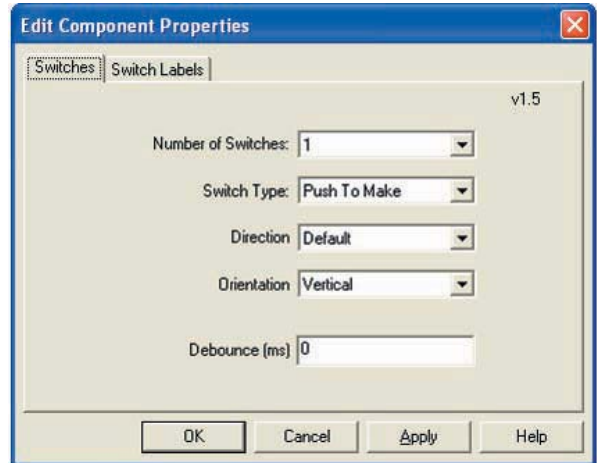
Fig.2. The initial Flowcode flowchart

● Now add the hardware. Click on the LED icon in the Components Toolbar, running just inside the Icon Toolbar. Also click on the switch icon. You now have two more items on the workspace.

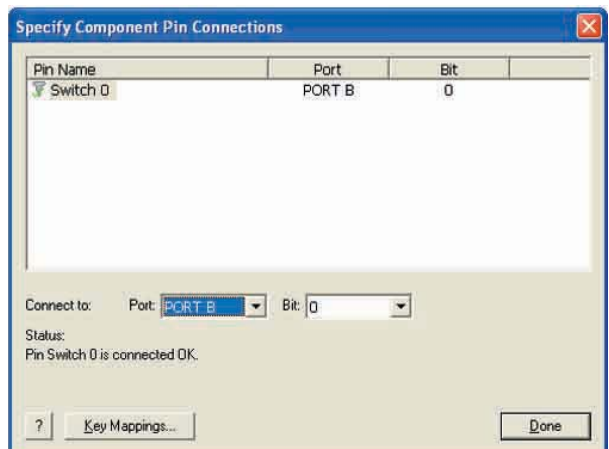
● Select the LED component in the workspace, click on the drop-down menu button and select the 'Component Connections' option. In the 'Connect to Port' drop-down menu, select Port D, (because the LEDs on the 44-pin Demo Board are connected to Port D.) The Connections dialogue box should now look like the following:



● Next, select the Switch component in the workspace. Click on the drop-down menu button, select the 'Properties' option, and set the number of switches to one.



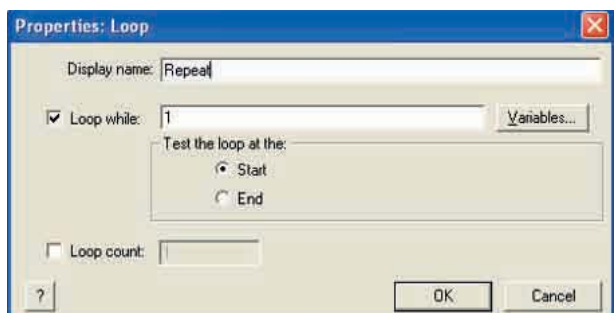
In the 'Connect to Port' drop-down menu, select Port B, (because the pushswitch on the 44-pin Demo Board is wired to bit 0 of Port B.) The Connections dialogue box should now resemble the following:



## Configuring the icons

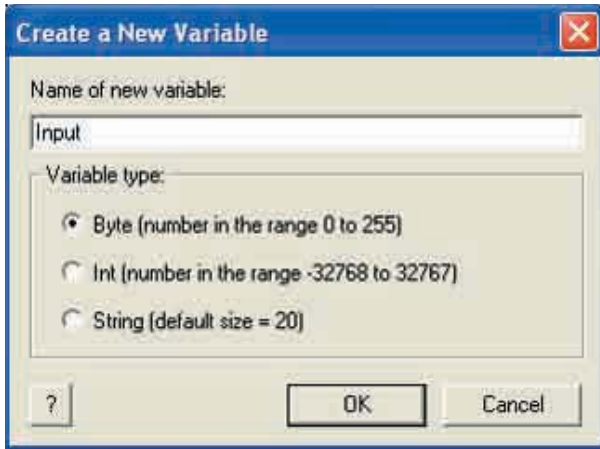
Now it is time to configure the icons that make up the Flowcode flowchart. To do this, double-click on each icon in turn and set up the configurations shown in the following diagrams. In the 'Display name' boxes, type the text from the corresponding element of the flowchart. This makes it easier to check your progress.

## The Loop icon

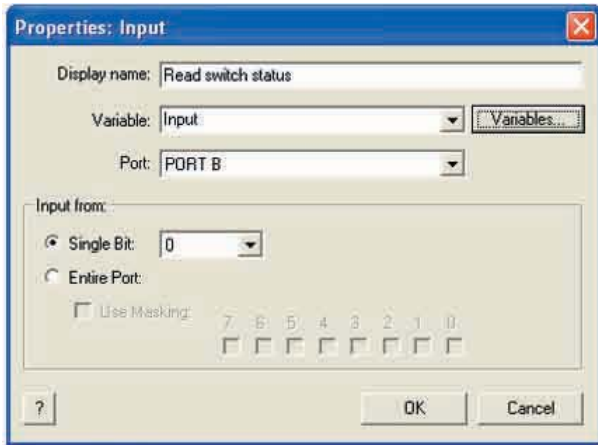


## The Input icon

First of all you need to create a variable to carry the switch information. Click on the 'Variables...' button to open the 'Variable Manager' window. Click on the 'Add New Variable' button, and type 'Input' as the name of the new variable, as shown in the next screenshot.

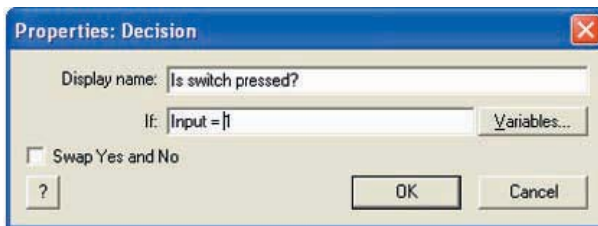


Then click 'OK'. Back in the Variable Manager window, click on 'Use Variable'. Back in the Input Properties window, select Port B, and 'Single Bit', so that the dialogue box now looks like:



## The Decision icon

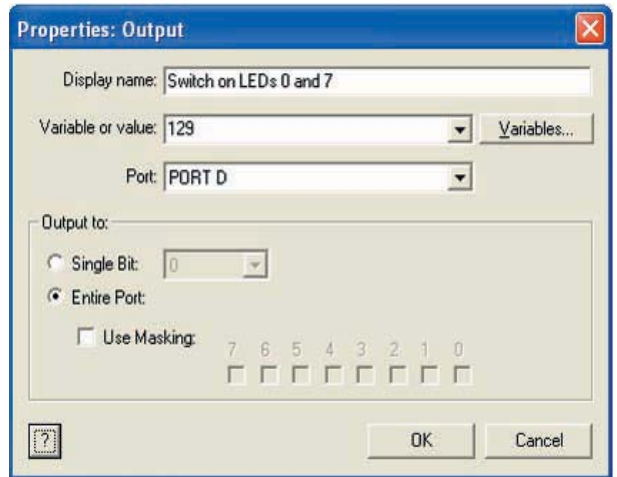
In the Properties dialogue box, click on the 'Variables' button and select the 'Input' variable for use. Type 'Is switch pressed' in the 'Display name' box. You should now have:



## 'No' Loop: The Output icon

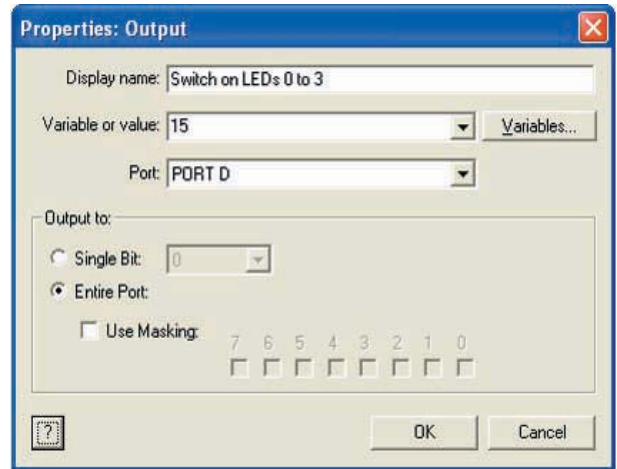
LED 7 is the most significant bit of the display, and so has a place value of 128 ( $= 2^7$ ).

LED 0 has a place value of 1 ( $= 2^0$ ). To make both light, you output a value of 129 to Port D, where the LEDs are connected.



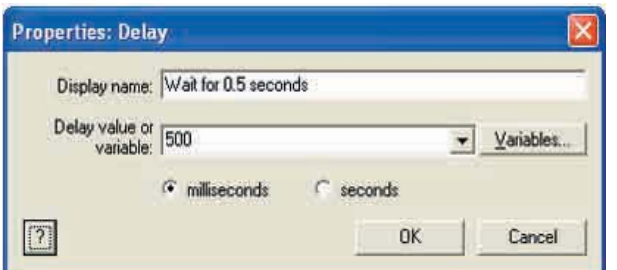
## 'Yes' Loop: First Output icon

The first four LEDs, 0 to 3, have place values of 1, 2, 4 and 8 ( $= 2^0, 2^1, 2^2$  and  $2^3$  respectively). You can light all four by outputting a value of 15 ( $= 1 + 2 + 4 + 8$ ) to Port D, as shown in the next screenshot.



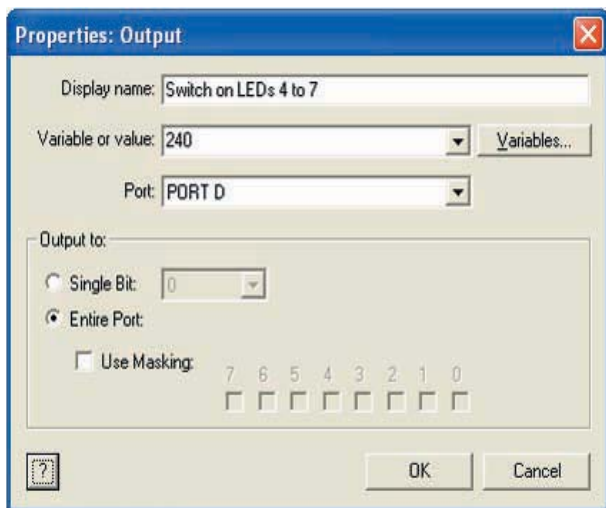
## Both Delay icons

The duration of the delay can be specified either in milliseconds or seconds. To create a half-second display, type 500 as the 'Delay value', with the 'milliseconds' button selected, since 500ms = 0.5s.



## Second Output icon

The LEDs 4 to 7 have place values of 16, 32, 64 and 128 ( $= 2^4, 2^5, 2^6$  and  $2^7$  respectively). To light all four, output a value of 240 ( $= 16 + 32 + 64 + 128$ ) to Port D, as shown in the next screenshot.



Now your Flowcode flowchart should look like Fig.3.

## Simulate the program

Flowcode allows you to check whether your program works by running an on-screen simulation. This can be done in two ways, simulating the full program, or stepping through the simulation, icon by icon. In both cases, two windows, the 'Variables' window, and the 'Call Stack' window, appear while the simulation runs. When you run the program simulation at full speed, the values in these windows are not updated as the program runs. If you slow down the full program simulation, or if you step through the program, then you will see the effect of each stage on the variables you created, shown in the 'Variables' window.

For now, look at the full simulation running at full speed. Click on the 'Run' command in the Menu toolbar. Choose the 'Go/Continue' option. The 'Variables' window, and the 'Call Stack' window appear, and LEDs 0 and 7 light. Click on the pushswitch icon, and the LEDs should flash on and off, with LEDs 0 to 3 alternating with LEDs 4 to 7. The program works!

(If it does not, then go back and check the configuration details given above. You could simulate the program step-by-step, by clicking on the 'Step Into' option in the 'Run' menu, and then using the F8 key to move to the next step.)

## Programming the PIC microcontroller

The groundwork is done – you have a working Flowcode program. The rest is easy! Here are instructions for getting your program into the chip:

- Connect the PICKIT 2 module to your computer using the USB cable.
- Plug the 44-pin Demo Board into PICKIT 2.
- Click on the 'Chip' command in the Menu toolbar, and select the 'Compile to Chip...' option. (You may be asked to save the Flowcode program – click on the 'Yes' button.)
- A 'Compiler Messages' window opens to show progress, as various elements are launched. The Target and Busy LEDs on PICKIT 2 light when the program is transferred. Finally, the 'Compiler Messages' window shows the word 'FINISHED', at which point you can click on the 'Close' button.
- Your program should now be running on the 44-pin Demo Board. LEDs 0 to 3 and 4 to 7 are alternately on then off. Press the pushswitch, and the LED pattern should change, with only LEDs 0 and 7 lit. That's what your Flowcode program said!
- Unseen by the user, the Flowcode programmer first compiled the graphical flowchart to a C file, and then compiled that to assembler. These can be viewed by C and assembler buffers either by using the 'View C' and 'View ASM' commands under the 'Chip' command on the Menu toolbar, or by opening the files *prog1name.c* and *prog1name.asm* in the folder where you saved the Flowcode program (where *prog1name* is the name you gave to that program.)

## Finally!

Although this article has shown in some detail how to build and configure Flowcode flowcharts, much of the functionality of Flowcode has not been mentioned. The ability to perform calculations, manipulate string variables, create customised macros and inject sections of C programming has not been mentioned, along with a host of other powerful features of the program.

Fortunately, the program comes equipped with a set of tutorials, accessed via the Help menu. Matrix Multimedia also offers a variety of supporting resources, available via their website [www.matrixmultimedia.com](http://www.matrixmultimedia.com). *EPE*

## Special offer

To start designing your new project, see the extra special offer from Microchip and *EPE* on the next page!

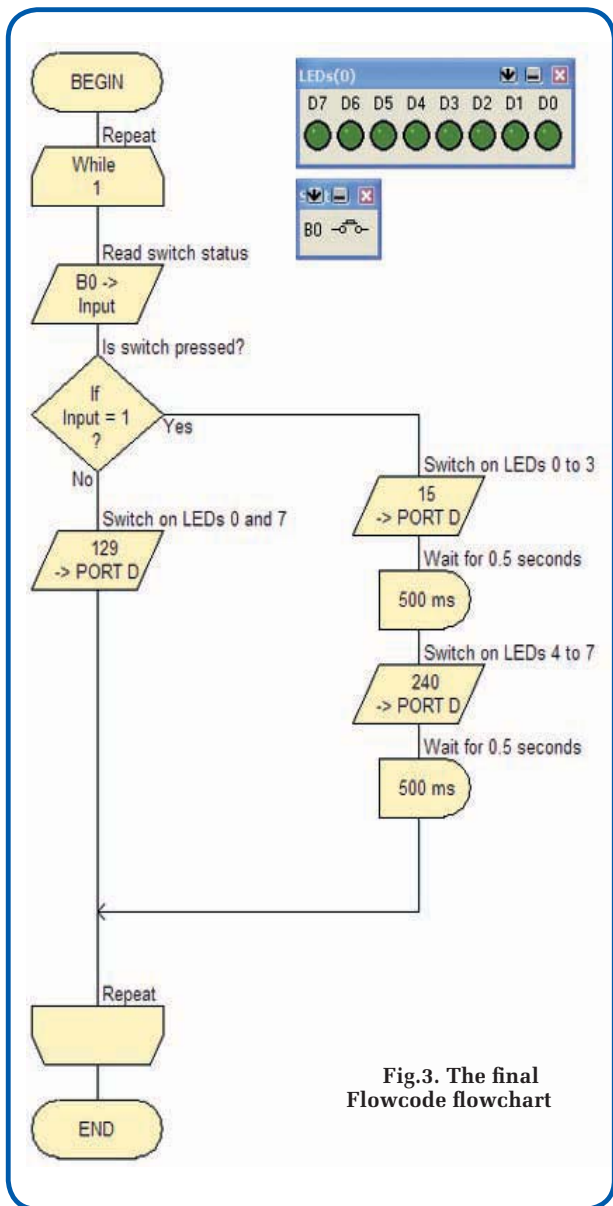


Fig.3. The final Flowcode flowchart