Flowcode is now available in several languages. Flowcharting facilitates a systems approach to electronics. The first benefit of using flow charts is that it allows teachers to use what school teachers in the UK call the ‘systems approach’ to teaching electronics. Using the systems approach students can build an electronic system of medium complexity from a number of sub-systems, and ‘glue’ these sub-systems together with flowchart software and a microcontroller. This approach has a number of benefits for students: it allows them to see how entire electronic systems are constructed, it gives them the relatively fast functional feedback they are used to, and developing medium complexity systems is highly motivating. It is true that students will not understand the detailed operation of all circuit elements in the system – but this can be addressed later.

The microcontroller as the core circuit element
This approach can be used over and over again: for example the operational theory and mathematics behind a potential divider circuit can be introduced to a student in the context of a simple thermistor-resistor temperature sensing circuit with a microcontroller making a decision on further actions under given temperature conditions. This approach to designing electronic systems mimics real designs: there are very few systems being designed today that do not have a microcontroller at their core. In fact a microcontroller has become a standard electronic component in much the same way that a 555 timer was a core component 10 years ago.

So the second benefit of using flow charts is that it allows all levels of students to use microcontrollers as a standard component – reflecting current industrial practice.

Programming is a key skill
Electronics is an industry driven by software. Every technician and engineer needs some form of programming skills. It is easy to understand that engineers need programming skills for modern electronic components like microcontrollers and CPLDs, but all technicians need some programming skills too: to enable them to drive the menu options on oscilloscopes, for operating automotive diagnostic systems, and often for the cafeteria drinks machines. Programming is now viewed as a key skill (along with numeracy and literacy) by many technical education institutions. Many educators find it difficult to decide what language to use for teaching fundamental programming skills. Whatever language is chosen, most teachers start by teaching the basic flow of a program using a flow chart. So there is an argument that it makes sense to use software based entirely on flow charts.

So the third reason for using flow chart programming software is that it teaches the fundamentals of programming itself, and allows students to concentrate on the strategy of programming and not get bogged down in the syntax of an advanced programming language like C. This applies to school students as well as University students.

Fast projects
One final reason for using flowcharts for programming microcontrollers is for project work. Most electronic courses involve some kind of project work. Many of these projects will need some kind of control system – for operating a front panel, for making run-time adjustments to a circuit etc. Using a flow chart for programming microcontrollers in projects allows students to concentrate more on the electronics part of the project than the software behind it. Using flow charts with microcontrollers effectively makes a project more functional, for very little extra effort. This applies to students who have learned C or assembly as much as low level students.