















## Further information

- **E-blocks2 hardware**

Datasheets for all E-blocks2 boards are available from the Matrix website: [www.matrixtsl.com](http://www.matrixtsl.com)

- CP2832 GSM Resource Files.zip  
also contains:
  - A PDF version of this document
  - Example files to accompany this document.

- **GSM Information**

This course is not designed around a specific GSM module, but one will be included in the package.

- **Software**

### **Flowcode Programming:**

Students should familiarise themselves with the basics of the language by reading the sections in the help file on adding icons, and components, and by working through the early tutorials to gain experience of how Flowcode works.

They can also use the “**An Introduction to Microcontroller Programming**” course. This is designed to take users from absolute basics through to quite advanced topics, and is a useful resource.

### **C or Assembly programming:**

For each of the more complex E-blocks2, a programming strategy guide is available on our web site. These give an outline of how each E-blocks2 can be programmed in C or Assembly code programmers.

If you are programming in C or Assembly, then you may benefit from one of our CD-ROM courses for programming PIC microcontrollers.

### ***Important!***

**Most of the exercises require the availability of a donor mobile phone to provide connectivity. Some of the programs dial a pre-programmed number when operating. A dummy number is programmed into the software before shipping, but this number must be changed to the number of the donor phone before compiling the software.**













































































































































































- Develop an alarm system that can send a message, including specific information, when triggered. Allow the alarm to be remotely reset.
- Use the LCD to display the progress of message reception and transmission.
- Make sure that the system can recover from any error conditions (unexpected responses from the modem etc.)

**Note:**

A solution to this exercise represents a complete remote telemetry application. Commands and data can be sent to the remote device in addition to the retrieval of data. In this trivial example the 'Status' command simply triggered the 'Ready' response. With appropriate string manipulation code, a range of commands and data can be used to control the remote application as well as retrieving data.

A possible application would be a burglar alarm system that could be activated, deactivated, and configured. The alarm would send a message containing details of any trigger events (zones triggered etc.) and allow remote resetting.