Theoretic/Practical   course on BASCOM AVR Programming.
Author: DAMINO Salvatore.

L C D - D I S P L A Y (2).

Message and Other Visualization   on LCD Display.

In this chapter we'll use the **High Level** instructions included in **BASCOM AVR** compiler that simplify the management of **LCD** display.

The **Example.021** program manages an interaction between user on **PC** and **Mini Module GMM AM08** mounted on **GMM TST3**, through a serial communication line in **RS 232**.

On **PC** monitor it is displayed a **Menu** where the user can select some operations. For each selected operation the program shows specifics indications and acquire relative data, inside the limits and bonds defined by the same operation.

The possible operations supported by the program are the following ones:

M   -> **Show message**

V   -> **Show variable**

P   -> **Position cursor**

C   -> **Define cursor style**

S   -> **Shift visualization**

E   -> **Erase visualization**

O   -> **Display on/off**

__Added Definitions:__
None

__Added Declarations:__
None

__Added Instructions:__
CURSOR ; SHIFTLCD ; DISPLAY

__Added Operators:__
None

Example program **2 1** of **BASCOM AVR** course.

It manages an alphanumeric display with **2 x 2 0** characters.

The program shows messages, numbers, variables and supplies commands to **GMM TST3** display. This display is **LCD** type back lighted by **LED**, it is provided of controller and it has **2** rows of **20** characters. The program completely manages it through the high level instructions dedicated to display, available in **BASCOM AVR**.

All the functionalities are established through a serial console provided of monitor and keyboard and it must communicate with a fixed physical protocol at **19.200 Baud**, **8 Bit x chr**, **1 Stop bit**, **No Parity**.

This console can be another system capable to support a serial **RS 2 3 2** communication. In order to simplify the use it can be used a **PC** provided of one **COMx** line, that execute a terminal emulation program as **HYPERTERMINAL** or the homonym modality provided by **BASCOM AVR** (see **IDE** Configuration).

The program works only when the **GMM AM08** is mounted on **Z1** socket of **GMM TST3**!!

Show Messages on LCD and Save them Inside Internal EEPROM

As previous program, also the **Example.022** program manages an interaction between user on **PC** and **Mini Module GMM AM08** mounted on **GMM TST3**, through a serial communication line in **RS 232**.

On **PC** monitor it is displayed a **Menu** where the user can select some operations. For each selected operation the program shows specifics indications and acquire relative data, inside the limits and bonds defined by the same operation.

Among the supported operations there is the possibility to save a message, long up to **40** characters, inside **EEPROM** of **Mini Module**. This message can be reloaded and shown on display, by choosing between **5** different messages.

Each message occupies **42** bytes of **EEPROM**: the first dedicated to a message saved indicator, the second dedicated to message length and the following **40** bytes dedicated to the maximum **40** characters of the message.

RS 232 Communication.

Example.022.   Save Strings on EEPROM and Shows them on LCD.

Added Definitions:
None

Added Declarations:
None

Added Instructions:
None

Added Operators:
None

Example program **2 2** of **BASCOM AVR** course.

It manages an alphanumeric display with **2 x 2 0** characters.

The program saves/load messages long **40** characters on/from **EEPROM** and i t shows them on **GMM TST3** display. This display is **LCD** type back lighted by **LED**, it is provided of controller and it has **2** rows of **20** characters.

The program completely manages it through the high level instructions dedicated to display, available in **BASCOM AVR**.

All the functionalities are established through a serial console provided of monitor and keyboard and it must communicate with a fixed physical protocol at **19.200 Baud**, **8 Bit x chr**, **1 Stop bit**, **No Parity**.

This console can be another system capable to support a serial **RS 232** communication. In order to simplify the use it can be used a **PC** provided of one **COMx** line, that execute a terminal emulation program as **HYPERTERMINAL** or the homonym modality provided by **BASCOM AVR** (see **IDE** Configuration).

The program works only when the **GMM AM08** is mounted on **Z1** socket of **GMM TST3**!!

88