

Course on BASCOM AVR - (10)

Theoretic/Practical course on BASCOM AVR Programming.

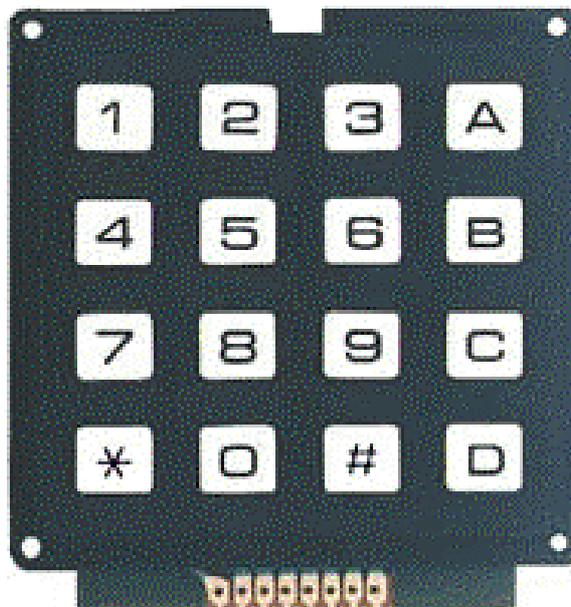
Author: DAMINO Salvatore.

KEYBOARDS (1).

In this chapter I introduce a new and important component: the **Keyboard**.

Through the keyboard it is possible to introduce some data into the system. They are composed by a variable number of keys (few or many) and they are available with different technologies and formats according with the final application field.

On **GMM TST3** it is used an inexpensive elastometric keyboard with **16** keys, in matrix **4x4** type.

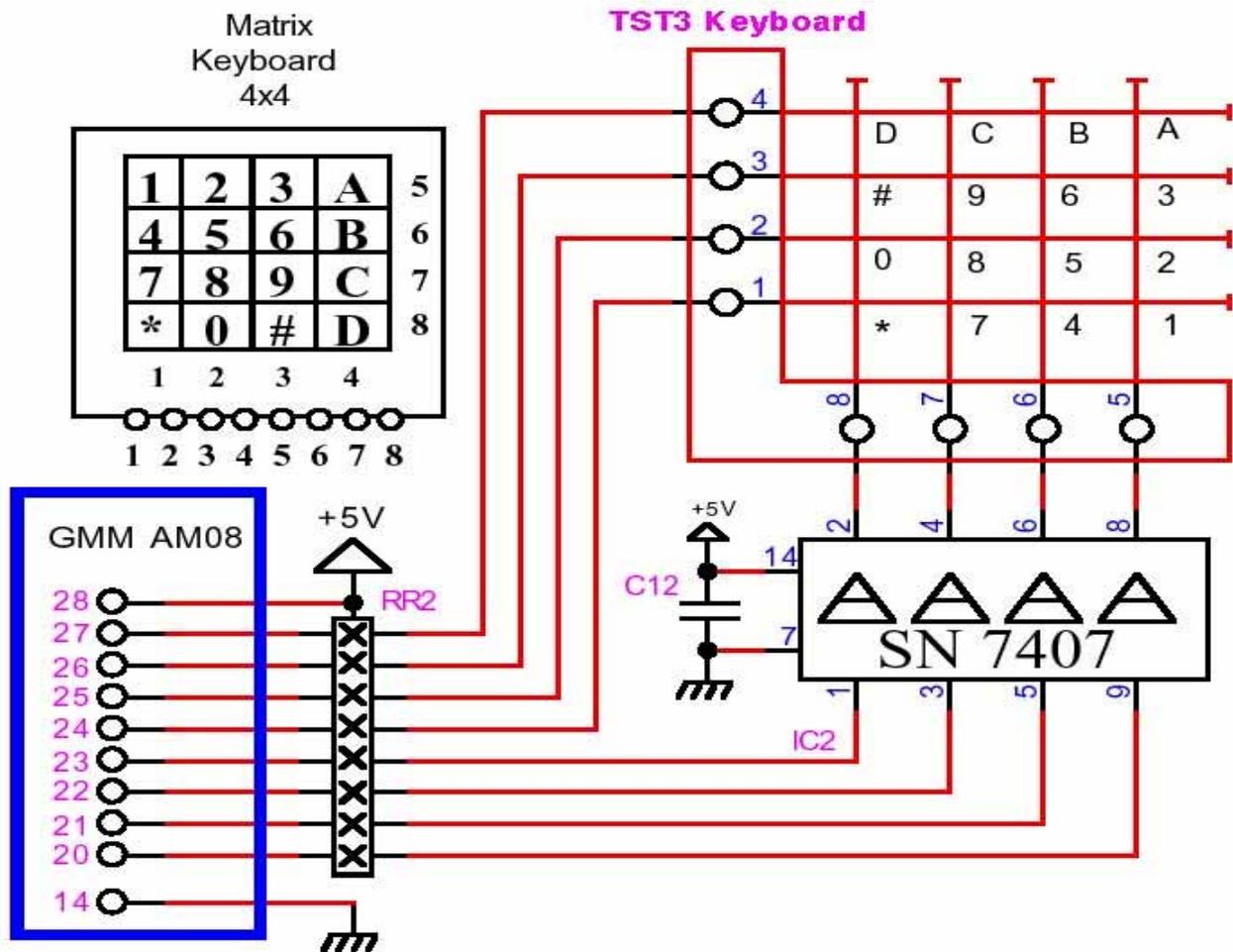


4x4 Keyboard.

On the keyboard are marked different symbols, digits and letters. In our condition there are **10** digits, the **Asteriks**, the **English Pound** and the letters **A, B, C, D**.

In the following two examples are described the possible modalities that can be used to acquire the various keys pressure.

In the first example is implemented a very rough modality that acquire only **4** keys at a time. These are those connected on the first column of the keyboard, previously selected. The described procedure can be repeated, by using the following columns, and so all the keys can be explored and checked.



Electric Diagram of 4x4 Keyboard.

The continuous read and check of the keys just described it is named **Scanning** and it is a typical operation used on each matrix keyboard.

The advantage of this modality is really obvious but, just to be really clear, below it is described with details.

If we had to acquire **16** keys by connecting an input lines to each key, then we must use **16 I/O** lines of our **CPU**. This is not a problem when we dispose of all these lines but if we have less lines, by using the **Scanning** modality, with only **8 I/O** lines we can solve the problem. The only disadvantage are some additional lines in the acquisition program but you save **8 I/O** lines that can be used for different functions.

Inside the **Example.017** we'll describe the **Scanning** program of a **4x4** matrix keyboard. The program doesn't take care of other typical problems of keyboard as rebounds and undefined acquisitions, that are not covered by using only this modality. If you use the keyboard in a sudden manner, for example by performing impulsive pressures, you can realize how this problems make the keyboard use unpleasant.

The solution to these problems is the argument of next chapter.

Example.016. Keyboard (4x4) Management. Only 4 Keys.

Added Definitions:

None

Added Declarations:

None

Added Instructions:

None

Added Operators:

None

Example Program.016 of **BASCOM AVR** course.

It manages **4** keys of a **4x4** matrix keyboard.

The program continuously acquires the state of **4** keys connected to first column of matrix keyboard available on **GMM TST3** and it transmits them on serial line. The acquisition is performed without **Debouncing**, without **Autorepeat** and without **Times Controls**.

The visualization of pressed keys is performed on a serial console provided of monitor 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!!**

Inside the program the row and column terms refers to electric diagram of matrix keyboard, not to its physical format!!

Example.017. Keyboard (4x4) Management. Transmit all Keys Pressed.

Added Definitions:

None

Added Declarations:

None

Added Instructions:

None

Added Operators:

None

Example Program.017 of **BASCOM AVR** course.

It manages **All** keys of a **4x4** matrix keyboard.

The program continuously acquires the state of **16** keys connected to first column of matrix keyboard available on **GMM TST3** and it transmits them on serial line. The acquisition is performed without **Debouncing**, without **Autorepeat** and without **Times Controls**.

The visualization of pressed keys is performed on a serial console provided of monitor 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!!**

Inside the program the row and column terms refers to electric diagram of matrix keyboard, not to its physical format!!