Course on BASCOM AVR - (6)

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

Simple Calculator for the 4 Operations.

These programs underline the high elaboration possibilities that can be achieved with the **BASCOM AVR** language, about mathematical operations.

Simple Calculator.

As elaboration example we illustrate how it can be realized a calculator capable to perform the **4** basic operations. It is clear that you can realize even more complex operations in a very simple way, in order to resolve expressions really more articulate.

Moreover these programs enlarge the user knowledge, in fact they introduce a list of new instructions as, for example, **INPUT**, **PRINTBIN**, **CASE**, etc.
Many of these new instructions mainly regards the console serial communication line. They allow to get data from external user, as the **INPUT** instruction does, or to show data for the user with the **PRINT** instructions.

Example.010. Calculator With 4 Operations (1)

Added Definitions:
None

Added Declarations:
Dim …... As Single

Added Instructions:
INPUT ; PRINTBIN ; SELECT CASE ; CASE ; END SELECT.

Added Operators:
- (subtraction) ; / (division)

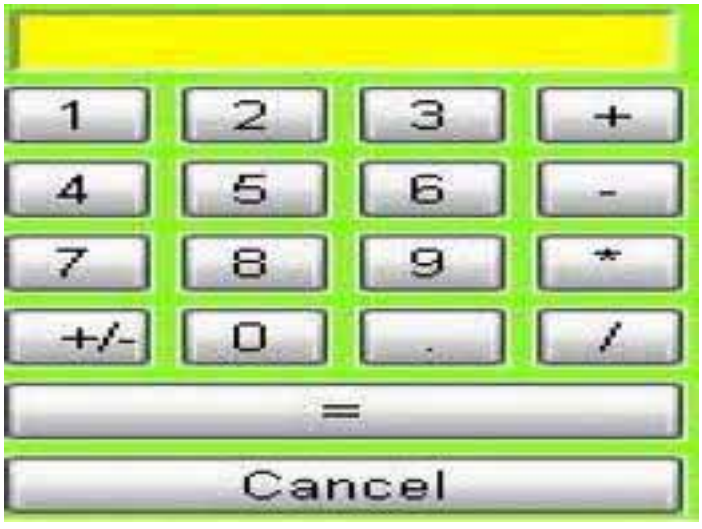
Example **Program.010** of **BASCOM AVR** course.

It manages a calculator that performs the **4** basic operations, on the **RS 232** serial communication line of **GMM AM08**.

The program requires a first operand, one operator and a second operand and then it shows the result of the performed operation. The operands can have sign and decimal point, up to **8** maximum significant digits.

The operands and operators are inserted through a serial console that shows also the result. The console must be provided of a keyboard and a 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 homonimous modality provided by **BASCOM AVR** (see **IDE Configuration**).



Example.011. Calculator With 4 Operations (2)

Added Definitions:
None

Added Declarations:
None

Added Instructions:
None

Added Operators:
None

Example **Program.011** of **BASCOM AVR** course.

It manages a calculator that performs the **4** basic operations, on the **RS 232** serial communication line of **GMM AM08**.

The program requires a first operand, one operator and a second operand and then it shows the result of the performed operation.
At this point it can be performed another operation on the obtained temporary result or terminate with **=**.

All the operands can have sign and decimal point, up to **8** maximum significant digits.

The operands and operators are inserted through a serial console that shows also the result. The console must be provided of a keyboard and a 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 homonimous modality provided by **BASCOM AVR** (see **IDE Configuration**).

Example.012. Calculator With 4 Operations (3)

Added Definitions:
None

Added Declarations:
Dim …... As String

Added Instructions:
DECLARE SUB ; SUB ; END SUB ; INSTR ; RIGHT ; LEFT ; VAL.

Added Operators:
None

Example **Program.012** of **BASCOM AVR** course.

It manages a calculator that performs the **4** basic operations, on the **RS 232** serial communication line of **GMM AM08**.

The program requires a formula composed by a first operand, one operator and a second operand on the same row and then it shows the result obtained from the formula. The operands can have sign and decimal point, up to **8** maximum significant digits.

The formula is inserted through a serial console that shows also the result. The console must be provided of a keyboard and a 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 homonimous modality provided by **BASCOM AVR** (see **IDE Configuration**).

In **Example.012** are introduced the **String** type **Variables** and some **Instructions** that use them.

It is strongly suggested that the reader get further information about this argument. At least I invite the reader to use the optimum on line help, in order to deeply understand the new used **instructions**.

The reader must remember that these **instructions** dedicated to strings management are frequently used both in current program and in following ones.