

# MB Additional String Util Libraries

- ‘- **Sub Procedure mem\_cpy(Dim addr1, addr2, n as word)**
  - ‘ Copies n bytes from the memory area starting at the address addr2 to addr1.
  - ‘ Ex. mem\_cpy(@data1, @data2, 4)
  
- ‘- **Sub Function mem\_Cmp(Dim addr1,addr2, n as word) as integer**
  - ‘ compares two memory areas starting at addresses addr1 and addr2 for n bytes
  - ‘ and return 0 if equal or difference if is not.
  - ‘ Ex. result16s = mem\_cmp(@data1, @data2, 4)
  
- ‘- **Sub Procedure mem\_set(Dim addr1 as word, Dim value as byte, Dim n as word)**
  - ‘ Fill n bytes from memory starting at address addr1 with value.
  - ‘ Ex. mem\_set(@data1, 255, 4)
  
- ‘- **Sub Function mem\_chr(Dim addr1 as word, Dim chr as byte, Dim n as word) as word**
  - ‘ Search for chr in first n bytes from memory starting at address addr1 and
  - ‘ return position index.
  - ‘ Ex. result16 = mem\_chr(@data1, “4”, 7)
  
- ‘- **Sub Function Str\_Len(Dim Byref data\_Str as String[4095]) as word**
  - ‘ Returns the length, in bytes, of the String data\_Str
  - ‘ Ex. result16 = Str\_Len(data1)
  
- ‘- **Sub Procedure Str\_Cut\_Chr(Dim Byref data\_Str as String[4095] , Dim chr\_ as byte)**
  - ‘ Remove all specified chr from front of the String data\_Str
  - ‘ Ex. Str\_Cut\_Chr(data1, “ “)
  
- ‘- **Sub Procedure Str\_Cat(Dim Byref data\_Str1, data\_Str2 as String[4095])**
  - ‘ Appends data\_Str2 to data\_Str1.
  - ‘ Ex. Str\_Cat(data1,data2)
  
- ‘- **Sub Function Str\_Cpy(Dim Byref data\_Str1, data\_Str2 as String[4095]) as word**
  - ‘ Copies the value of String data\_Str2 to the String data\_Str1
  - ‘ Return number of copied elements
  - ‘ Ex. Str\_Cpy(data1, data2)
  
- ‘- **Sub Function Str\_Chr(Dim Byref data\_Str as String[4095], Dim chr\_ as byte) as word**
  - ‘ Returns the position of the first character chr found in data\_Str
  - ‘ Ex. result16 = Str\_Chr(data1, “6”)
  
- ‘- **Sub Procedure Str\_Replace\_Chr(Dim Byref data\_Str as String[4095], Dim chr1, chr2 as byte)**
  - ‘ Replase all chr1 with chr2 in data\_Str
  - ‘ Ex. Str\_Replase\_Chr(data1,” “,”0”)

- ‘- **Sub Procedure Str\_Split(Dim Byref inst1,inst2 as string[4095], Dim n as word)**  
‘ Split inst1 in 2 strings after n char
- ‘- **Sub Procedure Str\_Insert\_Chr(Dim Byref inst1 as string[4095], Dim chr\_ as byte, Dim n as word)**  
‘ Insert Chr at position n
- ‘- **Sub Procedure Str\_AppendSuf(Dim Byref str\_ as string[4095], Dim ch as char)**  
‘ adds ch at the end of str\_
- ‘- **Sub Procedure Str\_AppendPre(Dim Byref str\_ as string[4095], Dim ch as char)**  
‘ inserts ch at the beginning of str\_
- ‘- **Sub Procedure Byte2StrWithZeros(Dim data\_in as byte, Dim Byref data\_Str as String[3])**  
‘ Convert byte value to String[3].  
‘ Ex. Byte2StrWithZeros(10, data1) ‘ data1 will be “010”
- ‘- **Sub Procedure Byte2Str(Dim data\_in as byte, Dim Byref data\_Str as String[3])**  
‘ Convert byte value to String, lenght of String is Variable from 1 to 3.  
‘ Ex. Byte2Str(10, data1) ‘ data1 will be “10”
- ‘- **Sub Procedure Short2StrWithZeros(Dim data\_in as short, Dim Byref data\_Str as String[4])**  
‘ Convert short value to String[4].  
‘ Ex. Short2StrWithZeros(-10, data1) ‘ data1 will be “-010”
- ‘- **Sub Procedure Short2Str(Dim data\_in as short, Dim Byref data\_Str as String[4])**  
‘ Convert short value to String, lenght of String is Variable from 1 to 4.  
‘ Ex. Short2Str(-10, data1) ‘ data1 will be “-10”
- ‘- **Sub Procedure Word2StrWithZeros(Dim data\_in as word, Dim Byref data\_Str as String[5])**  
‘ Convert word value to String[5].  
‘ Ex. Word2StrWithZeros(645, data1) ‘ data1 will be “00645”
- ‘- **Sub Procedure Word2Str(Dim data\_in as word, Dim Byref data\_Str as String[5])**  
‘ Convert word value to String, lenght of String is Variable from 1 to 5.  
‘ Ex. Word2Str(645, data1) ‘ data1 will be “645”
- ‘- **Sub Procedure Int2StrWithZeros(Dim data\_in as integer, Dim Byref data\_Str as String[6])**  
‘ Convert integer value to String[6].  
‘ Ex. Int2StrWithZeros(-645, data1) ‘ data1 will be “-00645”
- ‘- **Sub Procedure Int2Str(Dim data\_in as integer, Dim Byref data\_Str as String[6])**  
‘ Convert integer value to String, lenght of String is Variable from 1 to 6.  
‘ Ex. Int2Str(-645, data1) ‘ data1 will be “-645”

**‘- Sub Procedure LongWord2StrWithZeros(Dim data\_in as LongWord, Dim Byref data\_Str as String[10])**

‘ Convert LongWord value to String[10].

‘ Ex. LongWord2StrWithZeros(11645, data1) ‘ data1 will be “0000011645”

**‘- Sub Procedure LongWord2Str(Dim data\_in as LongWord, Dim Byref data\_Str as String[10])**

‘ Convert LongWord value to String, length of String is Variable from 1 to 10.

‘ Ex. LongWord2Str(11645, data1) ‘ data1 will be “11645”

**‘- Sub Procedure Long2StrWithZeros(Dim data\_in as longint, Dim Byref data\_Str as String[11])**

‘ Convert longint value to String[11].

‘ Ex. Long2StrWithZeros(-11645, data1) ‘ data1 will be “-0000011645”

**‘- Sub Procedure Long2Str(Dim data\_in as longint, Dim Byref data\_Str as String[10])**

‘ Convert longint value to String, length of String is Variable from 1 to 11.

‘ Ex. Long2Str(-11645, data1) ‘ data1 will be “-11645”

**‘- Sub Procedure Float2Str(Dim data\_in as Float, Dim Byref data\_Str as String[17] digits as byte)**

‘ Convert Float value to String, length of String is Variable from 1 to 17, max digits 4.

‘ Ex. Float2Str(-116.12345, data1, 0) ‘ data1 will be “-116”

‘ Ex. Float2Str(-116.12345, data1, 3) ‘ data1 will be “-116.123”

‘ Ex. Float2Str(-116.12345, data1, 4) ‘ data1 will be “-116.1234”

**‘- Sub Procedure Byte2Hex(Dim data\_hex as byte, Dim Byref hex as String[2])**

‘ Convert Byte to Hex

‘ Ex. Byte2Hex(255,data\_out)

**‘- Sub Function Hex2Byte(Dim Byref hex as String[2]) as byte**

‘ Convert Hex String format (must be String[2]) into Byte

‘ Ex. res\_8 = Hex2Byte(“AA”)

**‘- Sub Procedure Word2Hex(Dim data\_hex as word, Dim Byref hex as String[4])**

‘ Convert Word to Hex

‘ Ex. Word2Hex(1000,data\_out)

**‘- Sub Function Hex2Word(Dim Byref hex as String[4]) as word**

‘ Convert Hex String format (must be String[4]) into Word

‘ Ex. res\_16 = Hex2Worde(“AA55”)

**‘- Sub Function Str2Byte(Dim Byref byte\_in as String[3]) as byte**

‘ Convert Byte String format (must be String[1] to String[3]) into Byte

‘ Ex. res\_8 = Str2Byte(“6”)

‘ Ex. res\_8 = Str2Byte(“66”)

‘ Ex. res\_8 = Str2Byte(“ 66”)

‘ Ex. res\_8 = Str2Byte(“ 6”)

- ‘- **Sub Function Str2Word(Dim Byref word\_in as String[5]) as word**
  - ‘ Convert Word String format (must be String[1] to String[5]) into Word
  - ‘ Ex. res\_16 = Str2Word(“9”)
  - ‘ Ex. res\_16 = Str2Word(“ 669”)
  - ‘ Ex. res\_16 = Str2Word(“96”)
  - ‘ Ex. res\_16 = Str2Word(“ 669”)
  
- ‘- **Sub Function Str2LongWord(Dim Byref byte\_in as String[10]) as LongWord**
  - ‘ Convert LongWord String format (must be String[1] to String[10]) into LongWord
  - ‘ Ex. res\_32 = Str2LongWord(“1234567890”)
  
- ‘- **Sub Function Str2Short(Dim Byref byte\_in as String[4]) as short**
  - ‘ Convert Short String format (must be String[1] to String[4]) into short
  - ‘ Ex. res\_8s = Str2Short(“-6”)
  
- ‘- **Sub Function Str2Int(Dim Byref word\_in as String[6]) as integer**
  - ‘ Convert Word String format (must be String[1] to String[6]) into integer
  - ‘ Ex. res\_16s = Str2Int(“-12345”)
  
- ‘- **Sub Function Str2Long(Dim Byref byte\_in as String[11]) as Longint**
  - ‘ Convert Longint String format (must be String[1] to String[11]) into Longint
  - ‘ Ex. res\_32s = Str2dLong(“-1234567890”)
  
- ‘- **Sub Function Str2Float(Dim Byref byte\_in as String[17]) as Float**
  - ‘ Convert Float String format (must be String[1] to String[17]) into Float,
  - ‘ sign + 10bytes + “.” + 5 bytes
  - ‘ Ex. res\_float = Str2Float(“-123.12345”)
  
- ‘- **Sub Procedure IP2Str(Dim Byref user\_IPaddr as byte[4], Dim Byref Str\_out as String[15])**
  - ‘ Convert IP (byte[4] ) into String. Length of String is variable from 7 to 15.
  - ‘ Ex. IP2Str(IP\_address,data\_out)
  
- ‘- **Sub Procedure MAC2Str(Dim Byref MAC\_addr as byte[6], Dim Byref Str\_out as String[12])**
  - ‘ Convert MAC (byte[6] ) into String[12].
  - ‘ Ex. MAC2Str(MAC\_address, data\_out)
  
- ‘- **Sub Procedure Str2IP(Dim Byref Str\_in as String[15], Dim Byref user\_IPaddr as byte[4])**
  - ‘ Convert String (length of String must be from 7 to 15 and must include 3 “.”)
  - ‘ to byte[4]
  - ‘ Ex. Str2IP(“192.168.1.155”, IP\_address)
  
- ‘- **Sub Procedure Str2MAC(Dim Byref Str\_in as String[12], Dim Byref MAC\_addr as byte[6])**
  - ‘ Convert String (length of String must be 12) to byte[6]
  - ‘ Ex. Str2MAC(“00AA80FF457F”, MAC\_address)

- ‘- **Sub Procedure Str2IP\_(Dim Byref Str\_in as String[15], Dim IPaddr\_ as word)**
  - ‘ Convert String (length of String must be from 7 to 15 and must include 3 “.”)
  - ‘ to byte[4] and save to RAM starting at address IPaddr\_
  - ‘ Ex. Str2IP\_(“192.168.1.155”, @IP\_address)
  
- ‘- **Sub Procedure Str2MAC\_(Dim Byref Str\_in as String[12], Dim MAC\_addr\_ as word)**
  - ‘ Convert String (length of String must be 12) to byte[6]
  - ‘ and save to RAM starting at address MAC\_addr\_
  - ‘ Ex. Str2MAC\_(“00AA80FF457F”, @MAC\_address)
  
- ‘- **Sub Function Bcd2Dec(Dim number as byte) as byte**
  
- ‘- **Sub Function Bcd2Dec16(Dim number as word) as word**
  
- ‘- **Sub Function Dec2Bcd(Dim number as byte) as byte**
  
- ‘- **Sub Function Dec2Bcd16(Dim number as word) as word**
  
- ‘- **Sub Procedure PIC\_additional\_string\_library\_version(Dim Byref version as string[\$FF])**
  - ‘ return version of library
  - ‘ Ex. PIC\_additional\_string\_library\_version(data\_out)
  - ‘ data\_out will be “PIC\_A\_S\_L V4.0 12-May-2008”
  
- ‘- **Sub Procedure CpyFlash2Mem(Dim Faddr as LongWord, Dim Maddr\_,nb as word)**
  - ‘ Copies nb bytes from the Flash area starting at the address Faddr
  - ‘ to RAM address Maddr\_.
  - ‘ Ex. CpyFlash2Mem(@cd1,@data1,3)
  
- ‘- **Sub Function CpyFlashString2Mem(Dim Faddr as LongWord, Dim Maddr\_ as word) as word**
  - ‘ Copies String from the Flash area starting at the address Faddr to RAM area starting
  - ‘ at address Maddr\_.
  - ‘ Return number of copied elements
  - ‘ Ex. CpyFlashString2Mem(@cd1,@data1)
  
- ‘- **Sub Function CmpFlashWithMem(Dim Faddr as LongWord, Dim Maddr\_,nb as word) as integer**
  - ‘ Compare nb bytes from the Flash area starting at the address Faddr with RAM area
  - ‘ starting at address Maddr\_, return 0 if equal or difference if not .
  - ‘ Ex. result16s = CmpFlashWithMem(@cd1,@data1,3)
  
- ‘- **Sub Function CmpFlashStringWithMem(Dim Faddr as LongWord, Dim Maddr\_ as word) as integer**
  - ‘ Compare String from the Flash area starting at the address Faddr with RAM area
  - ‘ starting at address Maddr\_, return 0 if equal or difference if is not .
  - ‘ Ex. result16s = CmpFlashStringWithMem(@cd1,@data1)

**‘- Sub Function CmpFlashWithFlash(Dim Faddr1,Faddr2,nb as LongWord) as integer**

‘ Compare two Flash memory areas starting at addresses Faddr1 and Faddr2 for

‘ nb bytes, return 0 if equal or difference if not .

‘ Ex. result16s = CmpFlashWithFlash(@cd1,@cd2,5)

**Author : Florin Andrei Medrea, Software Version V3.0**

**Copyrights (c) 2008 - YO2LIO, All Rights Reserved**