

Modbus CRC uses the bit by bit method

Introduction:

This CRC (Cyclic Redundancy Check) is for the Modbus RTU protocol as used in RS232 and RS485 communications. The CRC is a 16 bit word that is calculated from all of the data in a message. It cannot correct any data corruption but it will indicate if a message has been corrupted in some way. This program uses the bit-by-bit method to calculate the CRC.

[Modbus, CRC](#)

PLEASE NOTE:

Although this is a Modbus CRC it is in no way restricted to be only used for the Modbus protocol. Any message structure consisting of a stream of 8-bit data bytes can utilise it. For example: Msg\$ = "The quick brown fox jumps over the lazy dog"

Acknowledgements:

This function was written by two members of The Back Shed forum:

TassyJim wrote the MMBasic version of the function. **Phil23** converted it to a string function to make it more user friendly.

I made a few tweaks to improve speed. Bill McKinley

An example a Modbus message (in Hex) could look like:

Slave Address	18
Function Code	03
Start Address (Hi)	0B
Start Address (Lo)	B9
Number of points (Hi)	00
Number of points (Lo)	01
CRC (Lo)	55
CRC (Hi)	C2

Note that the High/Low order of bytes in the CRC is the reverse of that used for the data. I don't know why.

In MMBasic without the CRC, that message would look like:

```
TxMsg$ = CHR$(&H18)+CHR$(&H03)+CHR$(&H0B)+CHR$(&HB9)+CHR$(&H00)+CHR$(&H01)
```

Add the CRC to the message by:

```
TxMsg$ = TxMsg$ + CrcStr(TxMsg$)
```

Performing a CRC check on a message that already includes a valid CRC will produce a 'CRC' of &h000 so: Check the received message (eg RxMsg) by:

```
IF CrcStr(RxMsg$) = CHR$(0) + CHR$(0) THEN ' CRC is OK
' Process the response
ELSE
' Bad CRC - do nothing - don't respond
END IF
```

Assumptions:

None

The Function:

```
FUNCTION CrcStr(a$) AS STRING
LOCAL ErrorWord% = &HFFFF, n, j, LSB AS INTEGER
FOR n = 1 TO LEN(A$)
ByteVal = ASC(MID$(a$, n, 1))
ErrorWord% = (ErrorWord% AND &HFFFF) XOR ASC(MID$(a$, n, 1))
FOR j = 1 TO 8
LSB = ErrorWord% AND &H0001
IF LSB = 1 THEN ErrorWord% = ErrorWord% - 1
ErrorWord% = ErrorWord% / 2
IF LSB = 1 THEN ErrorWord% = ErrorWord% XOR &HA001
NEXT j
NEXT n
CrcStr = CHR$(ErrorWord% AND &HFF) + CHR$(ErrorWord% >> 8)
END FUNCTION
```

From:
<http://fruitoftheshed.com/wiki/> - **FotS**

Permanent link:
http://fruitoftheshed.com/wiki/doku.php?id=mmbasic:modbus_crc_uses_the_bit_by_bit_method

Last update: **2024/02/24 17:35**

