

Validation of ChkSum-W32.exe

I. Requirement

FDA News Vol. 2 No. 23 page 4 states, "To determine that the hand-signed document is the same as the electronic record, the agency requires that the exact filename of the e-record, its size in bytes, the date and time of its creation and an authentication value, such as a check sum or unique mathematical value, are included on the handwritten copy provided to the FDA."

Requirement: Create a Windows based program that will calculate a check sum for a file.

Purpose: The check sum (along with the filename, size and creation time) will be used to verify that a file has not changed.

Scope: The check sum (along with the filename, size and creation time) will be used to associate a printed document with the file it was printed from.

II. Design

A program, ChkSum-W32.exe, will be developed in Microsoft Visual C++ .net 2003.

The program will calculate three authentication values:

1. A 2 character Logitudinal Redundancy Check (LRC) according to Modicon Modbus Protocol Reference Guide (PI-MBUS-300 Rev. J) Appendix C.
2. A 4 character Internet checksum according to RFC 1071.
3. An 8 character Cyclical Redundancy Check identical to the CRC32 value calculated by ZIP utilities (PKZIP or WinZip).

III. Construction

The check sum calculating program, ChkSum-W32.exe, was developed using Microsoft Visual C++ .net 2003. The C++ source code is in three files: CS-Win32.cpp, Crc32Static.cpp and PrintSupport.cpp. Printouts of these source code files are listed in the Appendix.

IV. Qualification

A tester shall execute steps 1 - 6 of this section.

An approver shall review the test documentation and approve the use of ChkSum-W32.exe to calculate checksums by signing at step 7 of this section.

1. LRC Verification

- a. Create a file of approximately 10 to 20 characters (Notepad, included with Windows, can create such a file). Record the filename and file-size (in XT Windows Explorer, right click the filename, select Properties, read size in bytes) here:

TestChecksum.txt, 30 bytes

- b. Using a program that will display the contents of a file in Hexadecimal units (one such program is Debug.exe, included with Microsoft Windows), record the Hexadecimal values, for all bytes of the file listed in step 1 a, here:

48 45 2D 35 31 31 20 44 75 74
79 20 69 73 20 31 2E 30 36 20
4D 4D 20 42 54 55 2F 48 72 2E

PK
4/13/04

c. Using the Calculator included with Windows (in Scientific mode), add the Hex values in step 1b, discard all carries above 1 byte (note that 1 byte = 2 characters in Hex), and record the sum here:

7D4, discard carries is D4

d. Subtract the value in step 1c from hex FF and record the remainder here:

FF - D4 = 2B

e. To the value in step 1d, add 1 and record here. This is the LRC calculated by hand.

2C

f. Record the LRC value that ChkSum-W32.exe calculates for the file listed in step 1a:

2C

g. If the hand calculated value in step 1e matches the automatically calculated value in step 1f, circle the word PASS. If not, circle the word FAIL.

2. Internet Checksum

a. Locate the "C" Implementation Example in section 4.1 of RFC 1071 (Note: RFC 1071 is attached to this document as Appendix 2).

b. Locate the function called GetFileChecksum() in the CS-Win32.cpp source code (attached to this document as Appendix 3).

c. For each line of code in the "C" Implementation Example, find and highlight the corresponding code in the GetFileChecksum function.

d. If the code in the RFC 1071 Example is the same as that in CS-Win32.cpp (except for defining sum as 64 bit), circle the word PASS. If not, circle the word FAIL.

e. Using the Hex values recorded in step 1b, pair adjacent Hex values (octets) to form 16 bit integers (if the number of octets is odd, add 00 at the end to fill out the pairs). Using the Calculator included with Windows, add the Hex values in step 1b as 16 bit integers, reversing the order of the octets to compensate for the Intel method of reversing the least and most significant bytes. Record the sum here:

3D503

f. Add the carry (beyond 16 bits) to the 16 bit result and record here (show the numbers added and the result).

3 + D503 = D506

*OK
4/13/04*

g. Subtract the value in step 2f from hex ~~FF~~^{FFF} (ones complement) and record the remainder here. This is the Internet Checksum calculated by hand.

FFF - D506 = 2AF9

h. Record the Internet Checksum value that ChkSum-W32.exe calculates for the file listed in step 1a:

2AF9

i. If the hand calculated value in step 2g matches the automatically calculated value in step 2h, circle the word PASS. If not, circle the word FAIL.

*OK
4/13/04*

3. CRC32 Verification

- a. Calculate the CRC32 value for three files (any files on your hard disk will do) using ChkSum-W32 and with a Zip utility. Record the filename and CRC32 values here:

Filename	CRC32 from Zip utility	CRC32 from ChkSum-W32.exe
<i>C5-win32.cpp</i>	<i>0x459349fb</i>	<i>459349fb</i>
<i>Printsupport.cpp</i>	<i>0xb5163eea</i>	<i>B5163EEA</i>
<i>Crc32-test.cpp</i>	<i>0x1fd7b635</i>	<i>1FD7B635</i>

- b. If the CRC32 value calculated by ChkSum-W32.exe is the same as the CRC32 value calculated by the Zip utility for each file, circle the word PASS. If not, circle the word FAIL.

4. Verify that the checksum for a file will change if even a single character of that file is changed.

- a. Select an existing text file or create a new text file with Microsoft Notepad (or other text editor). Calculate the LRC, Internet Checksum, and CRC32 for this file using ChkSum-W32.exe and record the checksums, along with the filename and file-size below.

filename	<i>TestChksum.txt</i>
File-size	<i>30 bytes</i>
LRC from ChkSum-W32	<i>2c</i>
Internet Checksum from ChkSum-W32	<i>2AF9</i>
CRC32 from ChkSum-W32	<i>345B55E0</i>

- b. Using Notepad (or other text editor), change 1 character of the file and save (the file-size should not change). Calculate the LRC, Internet Checksum, and CRC32 for this slightly changed file using ChkSum-W32.exe and record, along with the filename and file-size below

Filename (of slightly changed file)	<i>Testchksumr.txt</i>
File-size	<i>30 bytes</i>
LRC from ChkSum-W32	<i>0c</i>
Internet Checksum from ChkSum-W32	<i>0AF9</i>
CRC32 from ChkSum-W32	<i>E2BB79AB</i>

- c. If the LRC, Internet Checksum, and CRC32 have each changed from step 4a to step 4b, circle the word PASS. If any of these checksums has not changed, circle the word FAIL.

5. Document the version of ChkSum-W32.exe

Record the version number (from "About" menu), file-size, file "last write" date and time, and each of the checksums for the ChkSum-W32.exe file used in these tests.

filename	<i>chksum-w32.exe</i>
Version number	<i>0.06</i>
File size	<i>98304</i>
File "last write" date and time	<i>3/27/2004 16:41:04 UTC</i>
LRC	<i>52</i>
Internet Checksum	<i>2BD9</i>
CRC32	<i>B55D5599</i>

PK
4/13/04

6. Record tester

The tester shall record their printed name, signature, initials and date in the space below. Also initial and date each page of this document that has test results recorded, and attach a printout of the checksums calculated by CheckSum-W32.exe.

Printed Name	Signature	Initials	Date
<i>Pavl Kardos</i>	<i>Paul Kardos</i>	<i>PK</i>	<i>4/13/04</i>

7. Approval

The approver shall review all test steps above and the appendices for proper documentation. If all documentation is in order and each of the steps requiring a PASS has indeed passed, then the approver shall sign below indicating approval of the use of ChkSum-W32.exe for calculation of checksums.

Printed Name	Signature	Date
<i>Pavl Kardos</i>	<i>Paul Kardos</i>	<i>4/13/04</i>

V. Appendices

1. Modicon Modbus Protocol Reference Guide (PI-MBUS-300 Rev. J) Appendix C
2. RFC 1071 – Computing the Internet Checksum, September 1988.
3. Source code printout of CS-Win32.cpp
4. Source code printout of Cre32Static.cpp
5. Source code printout of PrintSupport.cpp

ChkSum-W32

File Help

File size, Date/time of Last Write, Checksum and Filename

```
30 bytes, 04/13/2004 17:02:14 UTC, LRC: 20, Testchksum.txt
30 bytes, 04/13/2004 17:02:14 UTC, I-CKSm: 2AF9, Testchksum.txt
6230 bytes, 01/11/2004 11:59:50 UTC, CRC32: 1FD78685, Crc32Static.cpp
24414 bytes, 03/27/2004 15:32:30 UTC, CRC32: 45949FB, CS-Win32.cpp
4540 bytes, 03/27/2004 13:28:12 UTC, CRC32: B5163EE0, PrintSupport.cpp
30 bytes, 04/13/2004 17:02:14 UTC, CRC32: 345055E0, Testchksum.txt
30 bytes, 04/13/2004 18:37:57 UTC, CRC32: E20049AB, Testchksumr.txt
30 bytes, 04/13/2004 18:37:57 UTC, I-CKSm: 0AF9, Testchksumr.txt
30 bytes, 04/13/2004 18:37:57 UTC, LRC: 00, Testchksumr.txt
98304 bytes, 03/27/2004 16:41:04 UTC, LRC: 52, ChkSum-W32.exe
98304 bytes, 03/27/2004 16:41:04 UTC, I-CKSm: 2BD9, ChkSum-W32.exe
98304 bytes, 03/27/2004 16:41:04 UTC, CRC32: B55D5599, ChkSum-W32.exe
```

PK
4/13/04