

Doc No.: AN-xxxx

Version: 1.2

Date: 04 January 2013

Subject: TextFileLoader

APPLICATION NOTE

1. What is the TextFileLoader?

The TextFileLoader is a solution to perform a fast file transfer from the PC to a Motion Coordinator.

There are two different versions of the TextFileLoader on the Motion Coordinators. The original version was developed on the MC2xx range of Motion Coordinators and requires configuration by the user. The MC4xx implementation is auto-configured by the MC4xx protocol.

1.1. MC2xx implementation

The TextFileLoader is implemented on the MC2xx LALA firmware version 1.6696 or greater; it is not available in the standard MC2xx build.

The MC2xx protocol must be started explicitly using the TEXT_FILE_LOADER command. This command never terminates, so it should be run as part of a single line program. The TEXT_FILE_LOADER command syntax is as follows

```
TEXT_FILE_LOADER[#<channel>,<fifo_name>,<fifo_size>,<vr_start>,<vr_size>,  
                <table_start>,<table_size>,<storage_area>,<protocol>
```

Where:

<channel> Numerical expression that specifies the communications channel to read/write

<fifo_name> Quoted string or VRSTRING that specifies the name of the FIFO file

<fifo_size> Numerical expression that specifies the size of the FIFO file to create. If the FIFO file already exists then this is ignored.

<vr_start> Numerical expression that specifies the start of the VR buffer:

VR(vr_buffer_base + 0) Status of the TEXT_FILE_LOADER state machine:

0 => Idle

1 => Receiving file name

2 => Receiving file data

4 => Aborting (received abort from PC)

6 => Dump mode idle (dump user request bit set)

7 => Dump mode dumping file name (dump user request bit set)

8 => Dump mode dumping file (dump user request bit set)

VR(vr_buffer_base + 1) User request, bitwise or between the following options:

1 => Abort (if a file is being received then the ABORT string is sent to the PC and the file waits until transfer has finished). This bit is cleared once the abort status has been set. If this bit is set again before the current file has finished then the transfer is aborted. This can cause a communications lockup.

2 => Dump (if a file is being received then the ABORT string is sent to the PC. All characters received are dumped until the dump mode is released.

4 => Back up the FIFO by the value of VR(vr_base + 2). This bit is cleared once the FIFO has been backed up.

8 => Clear the FIFO. This mode is released once the FIFO has been cleared by the FIFO read process.

16 => Freeze. No characters are processed until this mode is released by the user.

VR(vr_buffer_base + 2) File size. This value is 0 during the file transfer and is set to the number of bytes received when the file transfer terminates.

VR(vr_buffer_base + 3) Received filename as a VRSTRING.

<vr_size> Numerical expression that specifies the size of the VR buffer

<table_start> Numerical expression that specifies the start of the TABLE buffer that will be used to store the received data before writing to the SDCARD.

<table_size> Numerical expression that specifies the size of the TABLE buffer.

<storage_area> Numerical expression that specifies where to store the transferred data:

0 => nowhere

1 => FIFO

2 => SDCARD. The name of the file is the file name received from the PC.

<protocol> Protocol to be run

0 => MC2xx protocol

1 => Transparent protocol (file end is a timeout)

On the Euro209 port 3241 is redirected to channel 3.

1.2. Mc4xx implementation

The TextFileLoader is implemented on the MC4xx firmware version 2.0181 or greater. It is implemented as a special program, on the Motion Coordinator which is auto-started when a connection is detected on one of the TCP ports reserved for this purpose: port 3241 for transparent protocol and port 10001 for MC4xx protocol.

Once this special program has been started then it will be locked in the given protocol that caused it to auto-start. If the protocol must be changed then this special program must be stopped by performing one of the following: EX, cycle power, or STOP "TEXT_FILE_LOADER_PROGRAM".

Due to restrictions in the MC464 PRP-MIPS interface only 7 communications ports can be opened at any one time. These ports are selected by the IP_PROTOCOL_CONFIG parameter in the MC_CONFIG program on the MC464. The MC464 must be power cycled before changes to this parameter will be activated. This parameter is a bitwise or of the following values:

SERIAL_PORTS_FLAG = 0x0001,

```
TOKEN_TCP_FLAG           = 0x0002,
MODBUS_TCP_FLAG          = 0x0004,
ETHERNETIP_FLAG          = 0x0008,
OPENPLC_TCP_FLAG         = 0x0010,
HMI_FLAG                 = 0x0020,
DIRECT_TCP_FLAG          = 0x0040,
FINS_UDP_FLAG            = 0x0080,
FINS_TCP_FLAG            = 0x0100,
TFL_FLAG                 = 0x0200,
```

For example setting `IP_PROTOCOL_CONFIG=$77` will enable serial ports, PCMotion ActiveX, Modbus TCP, OpenPLC, Trio HMI and TextFileLoader transparent protocol. The default value for this parameter is `$23F`.

If transparent protocol is used then any program can be used to send the file to the controller but data cannot be compressed before transmission and the transferred data is always stored in the `TEXT_FILE_LOADER` file in internal RAM. If MC4xx protocol is used then the TextFileLoader protocol must be implemented, however the user has the flexibility to specify internal RAM, SDCARD or FIFO storage, and data compression options to improve the data transfer rate.

As an alternative Trio Motion Technology provides the `TextFileLoader.exe` application that implements all transfer protocols and also the `TextFileLoader` method in the ActiveX component.

The `TEXT_FILE_LOADER(1,0)` function can be called to read the status of the current file transfer. If this returns 0 then no transfer is active.

2. Examples

2.1. MC2xx implementation

All MC2xx transfers use port 3241. The storage location is specified by the `TEXT_FILE_LOADER` command itself.

2.1.1. Transparent transfer

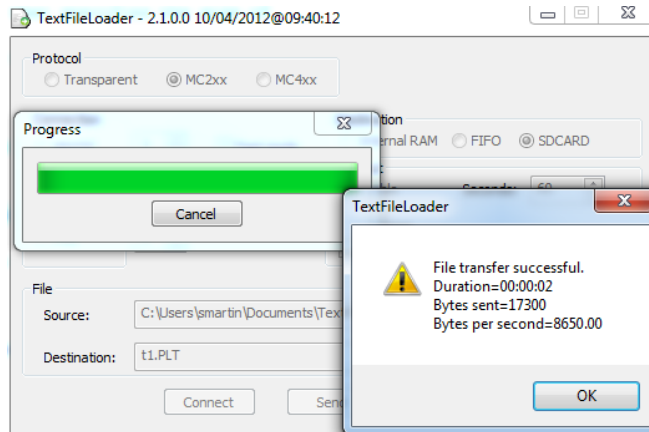
2.1.2. MC2xx protocol transfer

- Open MotionPerfectV2 and connect to the Euro209.
- Create a new program called “PC_TO_SD” on the Euro209 and type in the following commands.

```
TEXT_FILE_LOADER #3, "zzz", 256, 950, 50, 1000, 512, 2, 0
```

This will read data from channel 3 (Ethernet port 3241). Data will be stored on the SDCARD (the file name is sent by the TextFileLoader application). The FIFO file will be ignored, but the parameters must be specified.

- Run the PC_TO_SD program.
- Open TextFileLoader.exe. Select the correct IP address, MC2xx protocol and the “t1.PLT” file from 3.2.1 below. Click Connect and then Send.



- The file name is stored as a VRSTRING starting at VR(953).

```
>>? vrstring(953)
t1.PLT
```

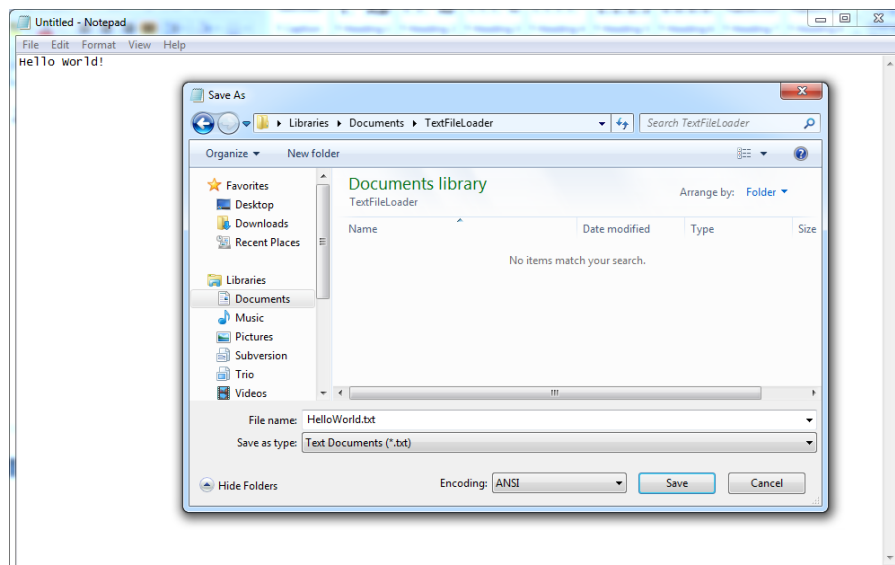
2.2. MC4xx implementation

2.2.1. Transparent transfer

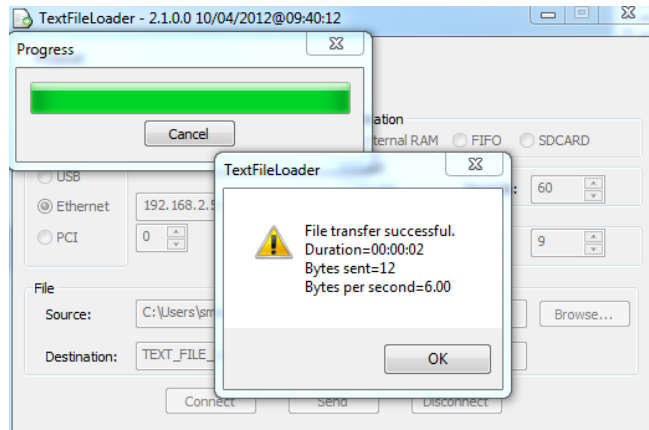
Port 3241 on the Motion Coordinator is used for transparent transfers. The PC application connects to this port sends the data down to the Motion Coordinator. The data will be stored in the internal RAM on the Motion Coordinator in a file called TEXT_FILE_LOADER. So that MotionPerfect will not try to synchronize the downloaded file with the current project, the file type is set to TEMP. TrioBASIC programs can access the downloaded file using the OPEN/GET/CLOSE commands.

2.2.1.1. Example using TextFileLoader.exe

- Create and save a text file on the PC.



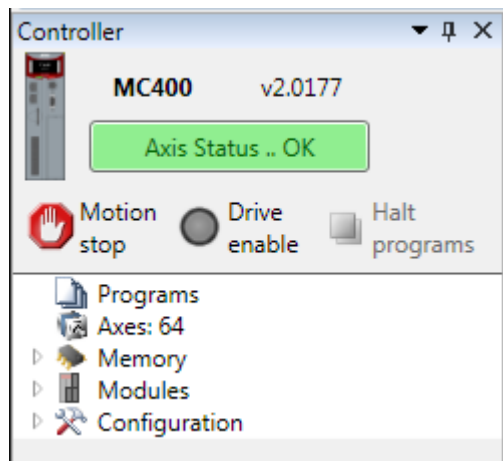
- Open TextFileLoader.exe. Select the correct IP address, no protocol and the file saved in the previous step. Click Connect and then Send.



- Connect to the Motion Coordinator with MotionPerfectV3, open the Channel 0 Terminal window and type in the DIR command.

```
>>dir
EPROM selected for power up
Memory available: 8388596
Selected program: TEXT_FILE_LOADER
Directory is UNLOCKED
Program          Source  Code   Run Type Code Type
-----
TEXT_FILE_LOADER  12     0 Manual Temp
OK
>>
```

As can be seen from the diagram below, the TEXT_FILE_LOADER file is not recognized by MotionPerfectV3 because the “Code Type” is “Temp”.



- Create a new program called “PRINT_FILE” on the Motion Coordinator and type in the following TrioBASIC commands.

```
OPEN #40 AS "TEXT_FILE_LOADER" FOR INPUT
WHILE KEY #40
  GET #40,c
  PRINT c[0];" ";CHR(c)
WEND
CLOSE #40
```

- This program opens the “TEXT_FILE_LOADER” file created by the transparent file transfer and prints its’ content to the Channel 0 Terminal window. In this case the output is as follows.

```

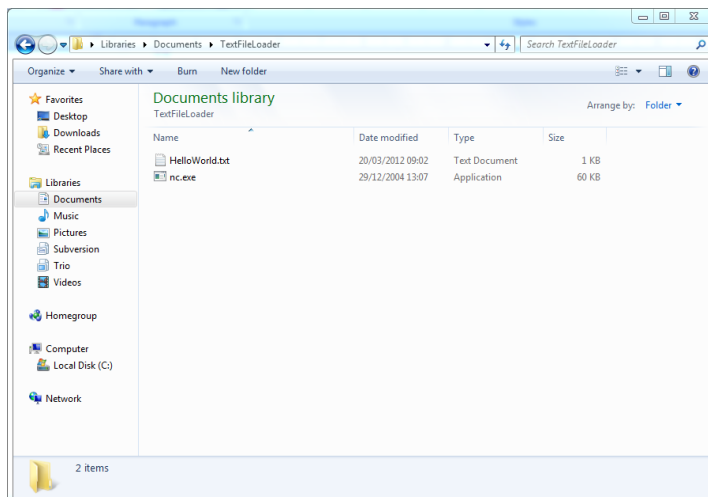
72 H
101 e
108 l
108 l
111 o
32
87 W
111 o
114 r
108 l
100 d
33 !

```

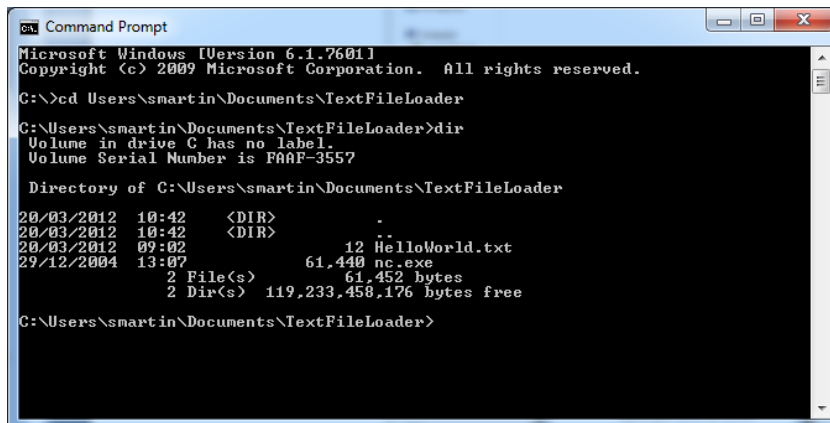
2.2.1.2. Example using netcat

Netcat is a simple networking utility which reads and writes data across network connections using the TCP/IP protocol. It's a wonderful tool for debugging all kinds of network problems. It allows you to read and write data over a network socket. At the time of writing this document a version for Windows can be downloaded from <http://joncraton.org/blog/46/netcat-for-windows>.

- Download netcat for Windows and copy nc.exe to the same directory the HelloWorld.txt file created in 2.2.1.1 above.



- Open a command line window and navigate to the folder.



- Send the file.

```

C:\> Command Prompt
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\> cd Users\smartin\Documents\TextFileLoader
C:\Users\smartin\Documents\TextFileLoader> dir
Volume in drive C has no label.
Volume Serial Number is FAAF-3557

Directory of C:\Users\smartin\Documents\TextFileLoader

20/03/2012  10:42    <DIR>          .
20/03/2012  10:42    <DIR>          ..
20/03/2012  09:02                12 HelloWorld.txt
29/12/2004  13:07                61,440 nc.exe
               2 File(s)              61,452 bytes
               2 Dir(s)          119,233,458,176 bytes free

C:\Users\smartin\Documents\TextFileLoader> nc -vv -w 1 127.0.0.1 3241 < HelloWorld.txt
smartin-alien [127.0.0.1] 3241 (?) open
net timeout
sent 12, rcvd 0: NOTSOCK
C:\Users\smartin\Documents\TextFileLoader>

```

- Use the same program created in 2.2.1.1 above to check the contents of the file.

2.2.2. MC4xx protocol transfer

Port 10001 on the Motion Coordinator is used for standard transfers. The PC application connects to this port sends the storage location, compression options and file data down to the Motion Coordinator.

3. Application Examples

3.1. MC2xx implementation

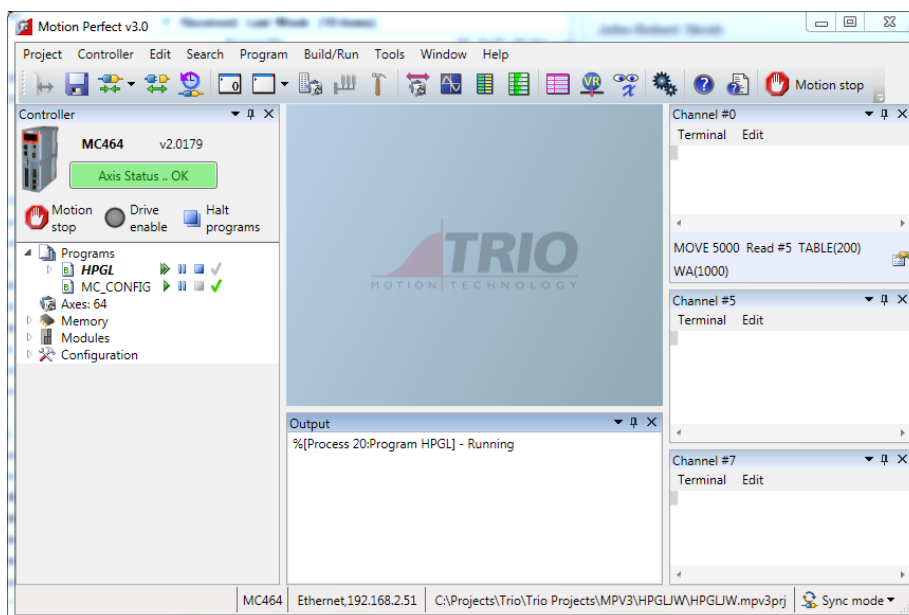
3.2. MC4xx implementation

3.2.1. HPGL parser



HPGL.zip

- Load the attached HPGL.BAS program onto the Motion Coordinator.
- Run the HPGL.BAS program.



4. Copy the attached T1.PLT file to the same directory as in the “Examples

4.1. MC2xx implementation

All MC2xx transfers use port 3241. The storage location is specified by the TEXT_FILE_LOADER command itself.

4.1.1. Transparent transfer

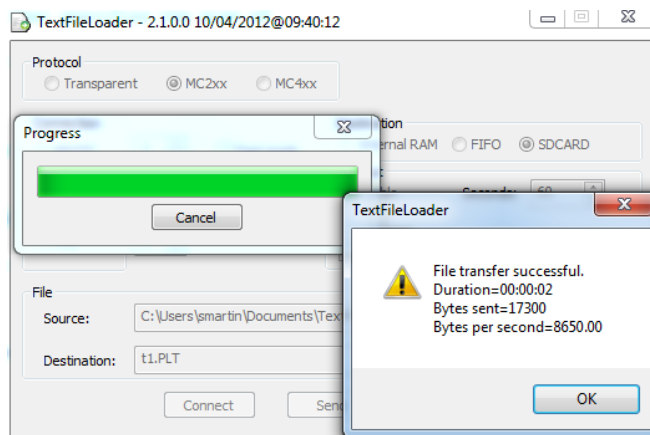
4.1.2. MC2xx protocol transfer

- Open MotionPerfectV2 and connect to the Euro209.
- Create a new program called “PC_TO_SD” on the Euro209 and type in the following commands.

```
TEXT_FILE_LOADER #3, "zzz", 256, 950, 50, 1000, 512, 2, 0
```

This will read data from channel 3 (Ethernet port 3241). Data will be stored on the SDCARD (the file name is sent by the TextFileLoader application). The FIFO file will be ignored, but the parameters must be specified.

- Run the PC_TO_SD program.
- Open TextFileLoader.exe. Select the correct IP address, MC2xx protocol and the “t1.PLT” file from 3.2.1 below. Click Connect and then Send.



- The file name is stored as a VRSTRING starting at VR(953).

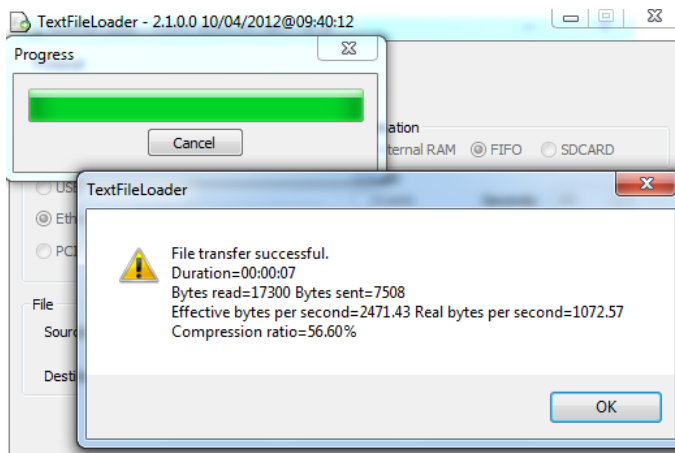
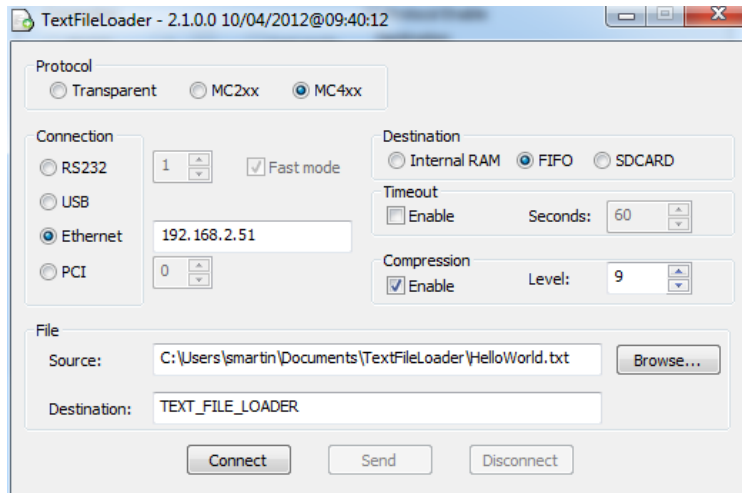
```
>>? vrstring(953)
t1.PLT
```

4.2. MC4xx implementation



t1.zip

- Transparent transfer” examples above.
- Transfer the T1.PLT file.



- The channel 7 window shows the parsed output from the HPGL program and the channel 5 window show the raw input read from the FIFO.

