User Manual for the
HE693CAN501

CANopen Master Module

Manual Version 4.3
Firmware version 2.73
25 Mar 2002

MAN0320-14

PREFACE

This manual explains how to use the Horner Electric's CANopen Module (HE693CAN501).

Information in this document is subject to change without notice and does not represent a commitment on the part of Horner APG, LLC.

For user manual updates, contact Horner Electric Advanced Products Group, Technical Support Division, at (317) 916-4274 or visit our website at www.heapg.com. or for European support Horner Electric at (++353) 21 4321 266.

LIMITED WARRANTY AND LIMITATION OF LIABILITY

Horner APG, LLC. ("HE-APG") warrants to the original purchaser that the CANopen Module manufactured by HE is free from defects in material and workmanship under normal use and service. The obligation of HE-APG under this warranty shall be limited to the repair or exchange of any part or parts which may prove defective under normal use and service within two (2) years from the date of manufacture or eighteen (18) months from the date of installation by the original purchaser whichever occurs first, such defect to be disclosed to the satisfaction of HE-APG after examination by HE-APG of the allegedly defective part or parts. THIS WARRANTY IS EXPRESSLY IN LIEU OF ALL OTHER WARRANTIES EXPRESSED OR IMPLIED INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE AND OF ALL OTHER OBLIGATIONS OR LIABILITIES AND HE-APG NEITHER ASSUMES, NOR AUTHORIZES ANY OTHER PERSON TO ASSUME FOR HE-APG, ANY OTHER LIABILITY IN CONNECTION WITH THE SALE OF THE Operator Station. THIS WARRANTY SHALL NOT APPLY TO THE Operator Station OR ANY PART THEREOF WHICH HAS BEEN SUBJECT TO ACCIDENT, NEGLIGENCE, ALTERATION, ABUSE, OR MISUSE. HE MAKES NO WARRANTY WHATSOEVER IN RESPECT TO ACCESSORIES OR PARTS NOT SUPPLIED BY HE. THE TERM "ORIGINAL PURCHASER", AS USED IN THIS WARRANTY, SHALL BE DEEMED TO MEAN THAT PERSON FOR WHOM THE Operator Station IS ORIGINALLY INSTALLED. THIS WARRANTY SHALL APPLY ONLY WITHIN THE BOUNDARIES OF THE CONTINENTAL UNITED STATES.

In no event, whether as a result of breach of contract, warranty, tort (including negligence) or otherwise, shall HE-APG or its suppliers be liable of any special, consequential, incidental or penal damages including, but not limited to, loss of profit or revenues, loss of use of the products or any associated equipment, damage to associated equipment, cost of capital, cost of substitute products, facilities, services or replacement power, down time costs, or claims of original purchaser's customers for such damages.

To obtain warranty service, return the product to your distributor with a description of the problem, proof of purchase, post paid, insured and in a suitable package.

ABOUT PROGRAMMING EXAMPLES

Any example programs and program segments in this manual or provided on accompanying diskettes are included solely for illustrative purposes. Due to the many variables and requirements associated with any particular installation, Horner Electric cannot assume responsibility or liability for actual use based on the examples and diagrams. It is the sole responsibility of the system designer utilizing the CANopen Module to appropriately design the end system, to appropriately integrate the CANopen Module and to make safety provisions for the end equipment as is usual and customary in industrial applications as defined in any codes or standards which apply.

Note: The programming examples shown in this manual are for illustrative purposes only. Proper machine operation is the sole responsibility of the system integrator.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

1.0. General

The HE693CAN501 is a Communications Interface Smart I/O Module for the GE Fanuc Series 90-30 PLC. Features include:

- Storage of configuration database in HE-693CAN501 module.
- Distribution of configuration database.
- Life-guarding function configurable through configuration database.
- HE693CAN501 self-configuration.
- Remote Slave reconfiguration.
- Start/Stop of one/several Slave node(s).
- Master/Slave switch.

1.1. System Requirements

The following equipment or software is required to use and or configure the CANOpen Master :

- Series 90-30 PLC
- LogicMaster Configuration/Programming Software for Series 90-30 PLC or VersaPro.
- Serial Programming Cable (HE693CBL232)
- Module programming Utility LoadMod.exe
- Software to produce *.rpt file, configuration tool or text editor. See HE693 CANopen Module Configuration Download Utility User’s Manual.

1.2. Physical Layout of HE693CAN501

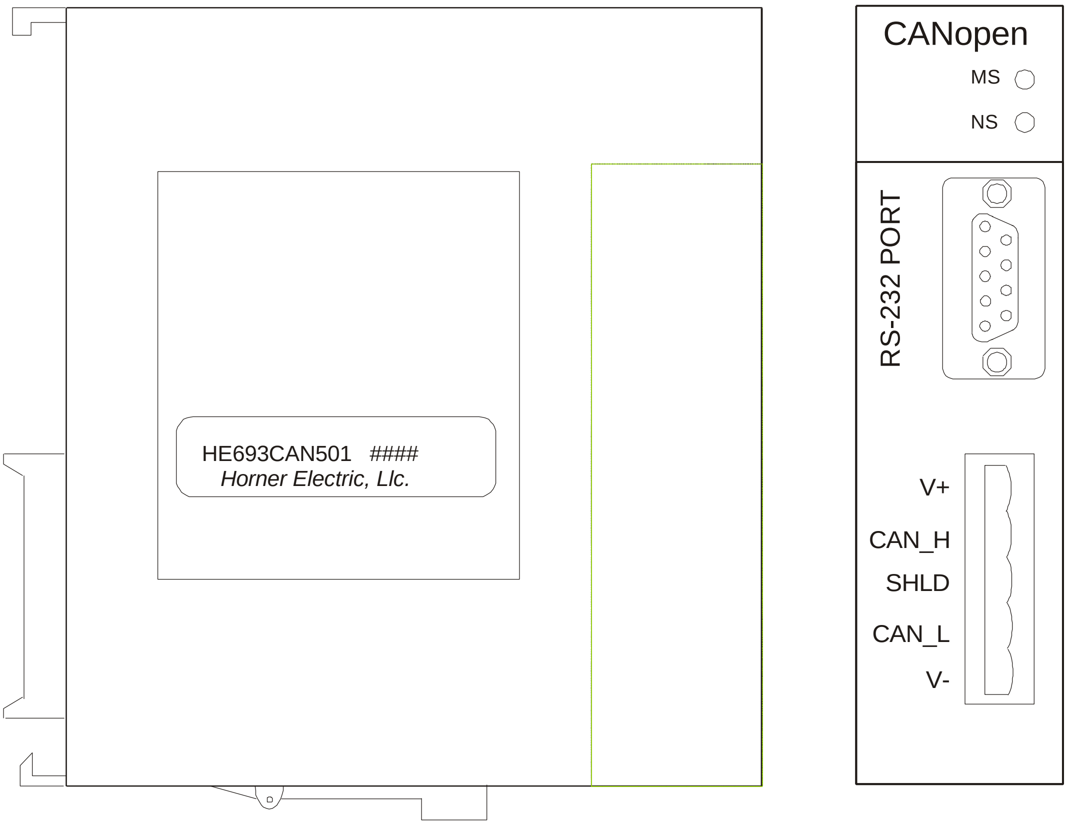

Figure 1.1 – HE693CAN501 Module (Front Cover and Side View)

1.3. RS-232 Connector

The RS-232 Service Port is used to download the configuration of the module and also as a service port to upgrade the firmware specific to the module. This port uses a standard 9-pin RS-232 connector.

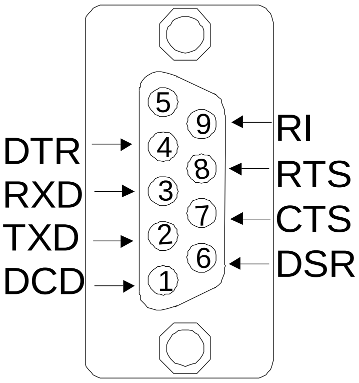

Figure 1.2 - Pin-out for the RS-232 connector.

CHAPTER 2 : OVERVIEW

In order to use the Series 90-30 CANopen Master both the module and the PLC must be configured.

There are two parts to the configuration :

- Some CAN settings for the module and the number of registers exchanged between the module and the PLC must be entered in the **3rd Party Configuration** area of the PLC.
- A **configuration file**, an formatted ASCII file containing objects according to the CANopen Object Dictionary specification must be downloaded to the module.

Both the module and the PLC must have knowledge about the details of the data to exchange. The following chapters explain the various steps to configure both the module and the PLC properly.

Two kinds of data are exchanged between the PLC and the Master and also between the master and the network: ***application data*** and ***status/command data***

Application data typically is analog or digital IO. Status and command data is the data the module provides to manage the CAN bus and the module itself.

To understand the following text you should have a basic knowledge about the CANopen terms and concepts.

2.1. Basic Settings

The parameters shown in Table 1. Must be entered in the Byte table of the PLC 3rd Party module configuration table. Each parameter has a specific purpose and the range of values chosen must be within the limits shown :

Parameter	Example	Available Values
Byte 1	00000001	Always 1
Byte 2	00000010	**CAN Open ID**
Byte 3	02H	**Can BAUD Rate**
Byte 4	00H	**SJW**
Byte 5	10H	**Network Initialization Time**)
Byte 6	10H	**Network Startup Delay**).
Byte 7-16	00H	Always 0

Table 1 - CAN Parameters

Byte 1 - **Always 1** if any other value used the communication between the PLC and the module will fail.

Byte 2 - **CANOpen ID**, This is the **NODE ID** for the module, each node on the network must have a unique ID between 1 and 31. The number should be entered in Binary.

Byte 3 - **CAN Baud Rate**, this is the CAN network baud rate, all nodes must be configured for the same Baud rate or the network will fail, only the following values are supported **0=125 kHz; 1=250 kHz; 2=500 kHz; 3=1 MHz).**

Byte 4 - **SJW**, this is the Sychronisation Jump Width which is to do with the point where CAN samples the incoming Data bits, **this must always be set to 0**.

Byte 5 - **Network Initialisation Time**, this time is the time delay from the start of the master sending Liefguard telegrams to the start of sending configuration SDO telegrams. The delay must be entered in **Hexadecimal** and the valid range is **1-65 seconds**.

Byte 6 - **Network Startup delay**, this time delay allows the slaves on the network time to boot-up, perform any internal initalisation sequences and enter pre-operational mode before the master sends NMT commands. It is entered in **decimal** and the value used muste be in the **range 0-255 seconds**.

Bytes 7-16 – **Not defined**, these parameters **must be zero**, any other value may prevent the module communicating with the PLC.

2.2. Configuring the Application Data Interface

Application data is transferred by PDOs via CAN to and from other CANopen nodes in the network. The module exchanges this data with the PLC via the Series 90-30 backplane.

The module supports two different PDO channels with different timing behaviour : Channel 1 for high speed data and Channel 2 for data with lower speed requirements.

The Series 90-30 PLC backplane interface offers two distinct methods of exchanging data with the CANopen Master module: ***PLC-initiated scanning*** and ***Module-initiated mail***.

PDO Channel 1 is driven by PLC-initiated scanning and PDO Channel 2 is driven by Module-initiated mail.

2.2.1. PLC-initiated scanning (PDO Channel 1)

The PLC-initiated scanning allows the transfer of a maxium of 124 Words in either direction. This corresponds to 31 PDOs with 4 words each. Configuration needs three steps:

- Two data areas must be assigned in the PLC, one for receive and one for transmit data.
- The module must be configured with the size (total number of words) of the transferred data.
- The required PDO's must be enabled.

The first two steps configure the data exchange between the PLC CPU and the module. The third step instructs the module to assign CAN messages (PDOs) to the data.

There is a fixed relationship between the position of the data in the assigned PLC data area and the PDO COB ID. The COB IDs are based on the "predefined connection set", PDO1, slave node ID 1-31 according to CANopen DS301.

2.2.2. Configure the PLC area

Enter the appropriate data in the PLC Hardware Configuration Screen of the PLC programming software (VersaPro or Logicmaster).

Use the same value for %AI and %AQ reference address.

Parameter	**Available Values**	
%I Ref Addr	%I0001	**variable**
%I Size	0	**variable**
%Q Ref Addr	%Q0001	**variable**
%Q Size	0	**variable**
%AI Ref Addr	%AI0001	**variable**
%AI Size	124	**variable**
%AQ Ref Addr	%AQ001	**variable**
%AQSize	124	**variable**
%R (in) Addr	%R0001	fixed
%R (in) Size	0	fixed
%R (out) Ref Addr	%R0001	fixed
%R (out) Size	0	fixed

Table 2 - PLC Register Allocation

The minimum applicable reference size can be calculated as 4 x highest slave address in the network. Higher values may be used but this increases the PLC's sweep time. Maximum size is 124 words which equates to 31 slaves.

2.3. Configure the module's data size

The number of %AQ and %AI registers declared in the 3rd Party configuration block must correspond to the number sent and received by the module. Some objects have been defined to allow the user to enter the number of registers required.

The default value is 124 Words for PDO1, this corrosponds to 31 nodes, the maximum supported by the master.

There are 9 latency levels supported by the master module, only levels 0 and 8 are CANOpen compliant, the remaining levels are not and are disabled by default. Each level corresponds to a different message throughput delay or latency, level 0 the PLC initiated Backplane scan being the fastest and level 8 the backplane mail driven method being the slowest. The number of registers required for each level must be defined seperately, the object dictionary entries are 4007..400F for level 0..8 respectively. These dictinary entries define the number of %AQ registers required in the PLC.

The following table gives the register types and assigned type numbers :

Reg Type (Register Type)			Permitted assignment / channel	
1:	AI	16bits		
2:	AQ	16bits	0-4	for 16bits registers
3:	I	1bit		
4:	Q	1bit	0-64	for 1bit registers
5:	No Type	No allocation		no assignment possible

Table 3 - HE-693CAN501 Register Types

:

For Transmit:

Object 4007,0	RegType	2	%AQ
Object 4007,1	RegNb	%AQ Reference Size	value from HWC

…

Object 400F,0	RegType	2	%AQ

The number of %AI registers is declared in a similar way but there is an object dictionary entry for each Node (slave) at each latency level.

For Receive:

Object 4010,0	RegType	2	%AI	Latency Level 0 Node 1
Object 4010,1	RegNb	%AI Reference Size	No. %AI for Node1	

...

Object 4018,0	RegType	2	%AI	Latency Level 8 Node 1
Object 4018,1	RegNb	%AI Reference Size	No. %AI for Node1	

.
.

Object 4126,0	RegType	2	%AI	Latency Level 8 Node 31
Object 4126,1	RegNb	%AI Reference Size	No.%AI for Node31	

Notes

1. The Object Dictionary entry numbers are always Hexadecimal numbers.
2. For CANOpen systems only object dictionary entries concerned with Latency levels 0 and 8 should be used.

2.4. MAPPING the PDO Data

Once the PLC registers have been assigned the data must be mapped to the CANOpen PDO's. Each PDO corresponds to four PLC words. Each PDO has a maxmum of eight bytes (this is the data limitation of a single CAN message), since the 9030 uses 16-bit registers this allows four assigned mapping objects, one for each PLC word. Each object can map 0 data, 8 bits (Low Byte) or 16 bits of the corresponding PLC word to the PDO. Each PLC register is automatically assigned an object dictionary entry, these objects are then used to define the PDO mapping.

Node No.	Dictionary Object & sub-Indices	Index in the block of Rx registers allocated to PDO channel 1 (per 16bits).	Dictionary Object & sub-Indices	Index for the block of Rx registers allocated to PDO channel 2 (per 16bits).
1	0x3000, 0..3	1..4	0x3000, 4..7	1..4
2	0x3040, 0..3	5..8..	0x3040, 4..7	5..8..
…				
30	0x3740, 0..3	117..120	0x3740, 4..7	117..120
31	0x3780, 0..3	120..124	0x3780, 4..7	120..124

Table 4 - Rx Dictionary Objects

There is a similar table for the transmit registers :

Node No.	Dictionary Object & sub-Indices	Index in the block of Tx registers allocated to PDO channel 1 (per 16bits).	Dictionary Object & sub-Indices	Index in the block of Tx registers allocated to PDO channel 2 (per 16bits).
1	0x2000, 0..3	1..4	0x2001, 60..63	1..4
2	0x2000, 4..7	5..8..	0x2002, 0..3	5..8..
…				
30	0x2001, 52..55	117..120	0x2003, 48..51	117..120
31	0x2001, 56..59	120..124	0x2003, 52..55	120..124

Table 5 -Tx Dictionary Objects

The HE-693CAN501 module supports up to 31 PDO telegrams for transmission and 31 PDO telegrams for reception over each of the two PDO channels.

The data fields of any of the 124 supported PDO telegrams (62 per channel) are mapped to the dictionary objects of Table 4. And 5 through the following dictionary objects:

PDO Channel 1 Receive Mapping objects :

Object	Receive PDO Ch1 Mapping 1	Default
1600, 0	NrOfMappedObjects	4
1600, 1	1st MappedObject	0
1600, 2	2nd MappedObject	0
1600, 3	3rd MappedObject	0
1600, 4	4th MappedObject	0

…

Object	Receive PDO Ch1 Mapping31	Default
161E, 0	NrOfMappedObjects	4
161E, 1	1st MappedObject	0
161E, 2	2nd MappedObject	0
161E, 3	3rd MappedObject	0
161E, 4	4th MappedObject	0

PDO Channel 1 Transmit Mapping objects :

Object	Transmit PDO Ch1 Mapping 1	Default
1A00, 0	NrOfMappedObjects	4
1A00, 1	1st MappedObject	0
1A00, 2	2nd MappedObject	0
1A00, 3	3rd MappedObject	0
1A00, 4	4th MappedObject	0

…

Object	Transmit PDO Ch1 Mapping31	Default
1A1E, 0	NrOfMappedObjects	4
1A1E, 1	1st MappedObject	0
1A1E, 2	2nd MappedObject	0
1A1E, 3	3rd MappedObject	0
1A1E, 4	4th MappedObject	0

2.5. Module-initiated mail (PDO channel 2)

There is no need to configure the PLC as to the number of registers required as this is done automatically by this transfer method, this reduces the configuration to the following two steps :

- Instruct the module about the size of the data and the referenced PLC addresses.
- Enable the required PDOs.

The first stage is to assign PLC register space, for the transmit data this is done using the object dictionary entry :

Object	TX PLC Mapping Table	Default
400F,0	Latency 8, Reg Type	5
400F,1	Latency 8, Reg No.	0

Channel 2 (PDO 2) Transmit PLC Register assignment object.

Assignement of registers for the receive side is done using the following dictionary objects :

Object	Node 1, RX PLC Map Table	Default
4018,0	Latency 8, Reg Type	5
4018,1	Latency 8, Reg No.	0

…

4126	Node 31, RX PLC Map Table	Default
4126,0	Latency 8, Reg Type	5
4126,1	Latency 8, Reg No.	0

Channel 2 (PDO 2) Receive PLC Register assignment object.

As with the PDO once the data area has been assigned the registers must be mapped to the PDO's, this is done using the following mapping objects :

Object	Receive PDO Ch2 Mapping 1	Default
1680,0	NrOfMappedObjects	4
1680,1	1st MappedObject	0
1680,2	2nd MappedObject	0
1680,3	3rd MappedObject	0
1680,4	4th MappedObject	0

…

Object	Receive PDO Ch2 Mapping 31	Default
169E,0	NrOfMappedObjects	4
169E,1	1st MappedObject	0
169E,2	2nd MappedObject	0
169E,3	3rd MappedObject	0
169E,4	4th MappedObject	0

PDO 2 Receive mapping object.

And

Object	Receive PDO Ch2 Mapping 1	Default
1680,0	NrOfMappedObjects	4
1680,1	1st MappedObject	0
1680,2	2nd MappedObject	0
1680,3	3rd MappedObject	0
1680,4	4th MappedObject	0

…

Object	Receive PDO Ch2 Mapping 31	Default
169E,0	NrOfMappedObjects	4
169E,1	1st MappedObject	0
169E,2	2nd MappedObject	0
169E,3	3rd MappedObject	0
169E,4	4th MappedObject	0

PDO 2 Receive mapping objects.

Object	Transmit PDO Ch2 Mapping 1	Default
1A80,0	NrOfMappedObjects	4
1A80,1	1st MappedObject	0
1A80,2	2nd MappedObject	0
1A80,3	3rd MappedObject	0
1A80,4	4th MappedObject	0

…

Object	Transmit PDO Ch2 Mapping 31	Default
1A9E,0	NrOfMappedObjects	4
1A9E,1	1st MappedObject	0
1A9E,2	2nd MappedObject	0
1A9E,3	3rd MappedObject	0
1A9E,4	4th MappedObject	0

PDO 2 Transmit mapping objects.

CHAPTER 3: HE693CAN501 MODULE NMT MASTER FUNCTIONS

3.0. General

The HE693CAN501 Module supports the following NMT Master functions :

- ***Master / Slave operation***
- ***Initial Boot-Up***
- ***Network configuration***
- ***Node Life-Guarding***
- ***Command channel***

3.1. Master / Slave operation

Master / slave operation is configurable at start-up. It is controlled by byte 5 of the PLC configuration window. If this memory location is set to 0, the HE693CAN501 Module runs in NMT Slave mode otherwise it runs in NMT Master mode and byte 5 is the ***Network Initialization Time*** encoded in seconds.

3.2. Initial Boot-up

This is broken into two parameters, Startup Delay and Network Initialization time. The startup delay is parameter 6 in the PLC configuration table and is the length of time for which the master delays attempting to configure slaves on the network. This delay gives slaves time to boot-up and complete any internal tests.
The Network Initialization Time (Byte 5 of the module configuration table) is the maximum start-up delay during configuration, by which time all NMT Slave Nodes on the network must have completed their internal initializations and configuration. When this delay expires, all the nodes known by the NMT Master to be in Pre-Operational mode are switched to Operational.

3.3. Network Configuration

When the module is in master mode and a valid configuration file has been loaded then it will act as the network master and configure all the slaves that have been declared in the configuration file. There may only be one configuration master on the network, if it required that another node act as the network configurator then do not load a configuration file.

3.4. SDO Time Out

Whe the master is downloading configuration objects to slave nodes the SDO Time out object may be used to allow slow devices time to process the configuration data before replying. This objet is part of the object dictionary for the master and is object 5000 SDO Time Out time. This object must be stored in the master section of the configuration file.

3.5. Lifeguarding

LifeGuarding operation is controlled by the configuration database stored in the Flash memory of the HE693CAN501 Module. For any Slave node in the range [1..31], LifeGuarding can be configured through one 16-bit unsigned integer and one 8-bit unsigned integer parameters: the ***Node Guard Time*** and the ***Node Life Time Factor*** respectively.

The ***Node Guard Time*** is the interval of time between two successive LifeGuard requests sent by the NMT Master Node to an NMT Slave Node. Each slave may have a different Node
Guard time.

The ***Node Life Time*** is the maximum amount of time the NMT Master Node will wait for a given NMT Slave Node to respond upon sending a LifeGuard request. The Node Life Time is the Node Guard Time multiplied by the Node Life Time Factor. A value of zero for a node’s Node Life Time object means that the node will neither be LifeGuarded nor switched to the Operational state when the NMT master starts the network.

Node Guard Times:	are initialized to the Slave Guard Times (Object located at 0x100C).
Node Life Times:	are initialized to the Node Life Time for Slaves whose **Life Time Factor** is greater than 0 (Object located at 0x100D) and to 0ms for all the other Slaves.

The Initial Boot-Up and Life-Guard sequencing performed by the HE693CAN501 Module is described as follows:

- At power-up, the HE693CAN501 Module initializes its Network Status Table to ***Unknown*** for all NMT Slave Nodes (See Status Table definition).

- ***Start-up delay*** executed here.

- ***NMT_Stop*** telegrams are sent to all the nodes for which the configuration data can be found in Flash memory. For those nodes, the field ***Node State*** of the Network Status Table is set to Pre-Operational.

- ***SDO telegrams*** are transmitted over the network to configure the nodes for which the configuration data can be found in Flash memory.

- ***Network Initialisation delay*** executed here.

- ***NMT_Start*** telegrams are sent to all the nodes known to be in the Pre-Operational state by the NMT Master provided that there were no SDO errors during configuration.

- The HE693CAN501Module starts sending LifeGuard request messages to all NMT Slave Nodes which have a non-zero **Life Time Factor**.

After a LifeGuard request message is sent to a NMT Slave Node, the HE693CAN501 module waits for the node's Node Life Time, for the node to respond with a LifeGuard data message containing the node's current status. If the node responds, the HE693CAN501 module updates its **Network Status Table** entry for the node, with the node's actual response time and its reported status (***Unknown, Pre-Operational*** or ***Operational***).
If Network Initialize Time has not yet expired, no error is logged for the nodes that do not respond to LifeGuarding. However, if Node Life-Guarding is operational, Network Initialize Time has not yet expired and a node responds and indicates that it is in a state other than Pre-Operational, the HE693CAN501 sends it an NMT start/stop command to place it in the Pre-Operational state.

If the Network Initialize Time has expired, the HE693CAN501 places all nodes which are known as Pre-Operational, in the Operational state by sending the NMT start command.
After the Network Initialize Time start-up period, LifeGuard requests sent to nodes are error checked. If a node does not respond, or if it responds with an unexpected state (i.e. not Operational), an appropriate error is logged.

For all three Life-Guard-related time values described above (Node Interval Time, Node Response Time and Network Initialization Time), the programmed value must be in the range [0.. 65530 ms] and must be an integral multiple of 10mS.

3.6. Network Status Table

The **Network Status Table** is located (by default) in the registers R1001 to R1124 of the PLC according to the following pattern for each node (8 bytes per node). It may be mapped to a different location in the PLC register set using the object dictionary mapping object 0x4004. This is a standard mapping object (see the CAN501 EDS file) :

R_x		R_{x+1}		R_{x+2}		R_{x+3}	
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Node State	**Last Life-Guard Result**	**Actual Response Time**		**Emergency Error Code**		**Error Register**	**SDO Status**
000-Unknown xxx-node states according to DS301	000-No Error 001-No Life-Guarding 254-Unexpected Response 255- No Response	[0..65535ms]		Error Code last transmitted by node in an Emergency telegram.			Status of last SDO

Table 9 - Network Status Table

In Table 9 replace R_x with: R_{status} for node 1, $R_{status + (NodeID – 1) * 4}$ for the node whose ID is NodeID.

R_{status} defaults to R1001. See chapter "Multiple Masters" for how to move it to a different address.

The Network Status Table includes also the status of the Master device.

Emergency Codes
The Network Status Table also shows the first bytes of the last slave emergency code. See CANopen DS301 "Emergency Object".

SDO Status
This byte reflects the status of the last parameter download to the relevant node. Zero means no error. A non-zero value means an error has occurred. Error codes are compacted error-class codes according to table 20 (SDO abort codes) of DS301, V4.0, page 9-26. Error class codes are in the high word of the 32-bit abort code. Compacted means that the word 0x0y is transformed to the byte xy.

Some common codes (compacted), for more details see DS301.
54 - time out (no response from the slave).
61 - access error (attempt to write a read-only value).
62 - object does not exist.
69 – sub-index does not exist.

Do not use the default locations of the status table for application data. They may be overwritten even if the table is remapped!

3.7. Command Channel
From the PLC to which the HE693CAN501 Module is linked, telegrams may be sent in order to reconfigure a Slave node, change its state or modify specific objects in a dictionary of another node. These commands can only be sent to nodes known to the HE-693CAN501 module (that is nodes with entries in the configuration file).

This is implemented through a set of specific registers in the PLC:

- One Command word.
- One Handshake word.
- SDO Data words.

By default the commands interface is disabled by setting the default length of the command word to zero. To enable the commands interface insert the following line into the RPT file:

```
4005,1 PLC command word       Var   UInt32      ReadWrite   0x07D00001
```

This enables the command channel with the command word (R_{cmd}) located at %R2001. See chapter "Multiple Masters" for how to move it to a different address.

R_{cmd}: Command Word	
Command Word MSB	Command Word LSB
0: all nodes (not available for the commands Reset Node and Reset Communication) 1-127: node to receive command	16: Configure Node 17: Send write-SDO telegram 128: Enter Pre-Operational Mode 129: Reset Node 130: Reset Communication 1: Start Remote Node 2: Stop Remote Node

Command Channel – Command word definition.

The control channel control bits (M_{hs}) defaults to %M1009. See chapter "Multiple Masters" for how to move it to a different address.

M_{hs}: Handshake Word		
M_{hs+2}..M_{hs+15}	**M_{hs+1}**	**M_{hs}**
Unused bits	693CAN501->PLC Flag	PLC->693CAN501 Flag

Command Channel – Handshake bits definition.

Commands are sent from the PLC to the HE-693CAN501 module as follows:

1. The PLC outputs the Command word to the HE-693CAN501.
2. The PLC sets the PLC-> HE-693CAN501 handshake flag.
3. The HE-693CAN501 reads the Command word, sends the command to the selected node and acknowledges by clearing the HE-693CAN501->PLC handshake flag.
4. The PLC clears the PLC-> HE-693CAN501 handshake flag.
5. The Command word is no longer valid (the PLC may now change it).
6. The HE-693CAN501 sets the HE-693CAN501->PLC handshake flag.

Command Descriptions:

16: Description:

- An NMT_Stop telegram is sent to the node specified in the MSB of R_{cmd}.
- SDO telegrams are sent to this node in order to reconfigure it with the data stored in Flash memory.
- An NMT_Start telegram is sent to the node specified in the MSB of R_{cmd}.

17: Description:

- An NMT_Stop telegram is sent to the node specified in the MSB of R_{cmd}.
- An SDO telegram whose Command Code is ‘Write’ and whose data are located in R_{SDO}-R_{SDO+3} is sent to this node.
- An NMT_Start telegram is sent to the node specified in the MSB of R_{cmd}.

Telegram Fields							
R_{SDO}		**R_{SDO+1}**		**R_{SDO+2}**		**R_{SDO+3}**	
MSB	LSB	MSB	LSB	**Data: MSW**		**Data: LSW**	
Index		**Sub-Index**	**Data Type:**				
			0x01: UInt8	MSB			
			0x02: UInt16	MSB	LSB		
			0x04: UInt32	MSB	LSB	MSB	LSB

R_{SDO} defaults to %R1996. See chapter "Multiple Masters" for how to move it to a different address.

3.8. Multiple Masters

To use more than one master module in a rack, object dictionary entries 4004 and 4005 are available to move the addresses of Status, Command, Handshake and SDO data to different locations. It is up to the ladder logic to keep track of data going to and from each module, the master modules act independently of each other and can only communicate via the ladder. The masters must be on physically different networks. The following objects must be re-mapped to prevent data or control overlapping :

Map	*Name*	*Object*	*Default value*	
PLC Status map	Network Status	4004	0x03E80080	size=128, type=%R, offset=1001
PLC Control map	Command Word	4005,1	0x07D00000	size=0, type=%R, offset=2001
PLC Control map	Handshake Word	4005,2	0x03F00410	size=16, type=%M, offset=1009
PLC Control map	SDO Data	4005,3	0x07CB0004	size=4, type=%R, offset=1996

Each entry has the same format:

Byte 0	Byte 1	Byte 2	Byte 3
Size	Register Type	Starting Offset - 1	

Setting the size of the Command word to zero results in disabling the commands interface.

Register Type is encoded as follows:

coding	Register Type
00	%R
01	%AI
02	%AQ
03	%T
04	%M
05	%I
06	%Q
07	No Type

Register Type Enumeration

Do not change the register type.

Example:
the following object definition is equal to the defaults, except that the length of the command word is set to 1 to enable the commands interface:

```
4004,0 PLC Status Map        Var   UInt32      ReadWrite   0x03E80080
4005,1 PLC command word      Var   UInt32      ReadWrite   0x07D00001
4005,2 PLC Handshake word    Var   UInt32      ReadWrite   0x03F00410
4005,3 PLC SDO data words    Var   UInt32      ReadWrite   0x07CB0004
```

CHAPTER 4: HE693CAN501 MODULE TIMING CONSIDERATIONS

4.0. General

There are two issues which must be considered when putting together a system using the HE693CAN501 Module these are as follows :

- ***Data delay***
- ***Data Overload.***

4.1. Data delay

Depending upon the configuration data delay may or may not affect the final system performance. There are two methods of communiction between the CPU module and the CANOpen Master module :

- ***Direct backplane access (limited to 127 Words in and 127 words out)***
- ***Backplane mail (any number of register).***

Direct backplane access is the faster of the two systems and is supported by the Master module (it is the preferred method of data transfer). Exchanging 124 words using direct backplane transfer takes ≈ 4ms. As well as this delay there is a finite time to take the message from the CAN controller and put it on thebackplane via the data queues. The delay for this varies with the number of nodes configured and whether life-guarding is required or not. Data delay will not cause the loss of data, it may affect a control loop where the PLC is using the CANOpen master within to send commands and receive feedback. The following two equations may be used to indicate approimate delay times for non life-guarding and life-guarded systems :

No Life-guarding :

Delay (Sec) = (No. of words * 0.00003225) + 0.000250

With Life-guarding :

Delay (Sec) = (No. of words * 0.00003225) + 0.000300

Where :
No. of words = the number of words defined in the configuration to transfer between the CPU and the Master Modules.

4.2. Data Overload

Data Overload is caused by the Master module not being able to keep up with the incoming CAN messages. Both the receive and transmit sections have message queues to prevent data loss. The direct backplane access and backplane mail sub-systems have the highest priority in the system, servicing the CAN port is next. Once data has been received it is queued until the relevent prioity (lower) loop has time to process it. If data is put into the queue continueously faster than it is removed eventually the queue will fill and messages will be lost. Using queues allows short high volume traffic without data loss. The following calculations are simplifications of calculations used in current queuing theory and may be used to determine whether data overloading is likely to occur with the chosen number of nodes and message rates :

Definitions :
λ - input message rate.
μ - service rate. (approx. 150µS for the CANOpen Master)
Pq – Probability of there being a message in the queue.
Ps – Probablitiy of the service being busy.
N – Number of messages in the queue.

$Pq = \lambda / (\mu - \lambda)$ and $Ps = \lambda / \mu$

$N = (Pq – Ps) * 1000$

It is important to express the input message and the service rates in seconds, the input message rate can be calculated as follows :

Input Message rate = ((Number of Nodes @ Time 1) in secs) + ((Number of Nodes @ Time 2) in secs) +…((Number of Nodes @ Time N) in secs).

Where the times are the cyclic times between messages. This allows an average loading.

The service rate is fixed at 6666.66 messages per second. This is a feature of the card and is an average value, it varies depending upon other events at any particular instant in time.

Finally N the number of messages must never exceed 200, this is a fixed value in the unit. The above calculations are average figures, they do not allow for worst case timing varations or bursts of higher density traffic, to allow for this ensure that N < 150 for normal operation.

CHAPTER 5: DEFINITION OF LED STATUS INDICATORS

The two Leds located in the upper area on the front bezel of the HE693CAN501 Module inform the user of the node's current state. Please check Table 3 for details:

Table 3 – Module States

Description	**NS Led Status**	**MS Led Status**
Firmware upgrade: After power-up, the NS Red Led is ON for about 5s. During this period of time, a new firmware can be downloaded into the HE693CAN501 module. Do not load *.RPT files during this period.	RED	N/A
Configuration transfer: * While configuration data are transferred over a serial link by the LoadMod software, the MS Green Led flashes regularly.	N/A	GREEN FLASH
Network configuration: While configuration data are dispatched over the CAN network, the MS Red Led flashes regularly.	N/A	RED FLASH
Operational State: As soon as the network configuration process has been completed and the Network Initialization Time expires, a node configured to be a NMT Master should automatically switch to the Operational State. The MS Green Led lights up to show this state.	N/A	GREEN
Pre-Operational State:	N/A	OFF
Network failure: A CAN bus recovery sequence is triggered.	RED	N/A
Network Operating: Transmission / Reception.	RED FLASH	GREEN

CHAPTER 6: HE693CAN501 CONFIGURATION DATA

6.0. Configure Object Dictionaries

The configuration data for each node must be preceded by the node ID and node Name, the data consists of dictionary entries and object data and is formatted as follows:

```
aaaa,bb        cccccccc        Var     dddddd ReadWrite     eeeeeeeee
```

aaaa	Is an **index** of a dictionary entry coded in hexadecimal.
bb	Is a **sub-index** of a dictionary entry coded in hexadecimal. Some dictionary entries do not have sub-indices.
cccccccc	Is the **name** of the dictionary object located at aaaa,bb. Any string up to 40 ASCII characters long describing the object function (non control characters).
dddddd	Is the **data type** of the object located at aaaa,bb. Valid values are: 'UInt8', 'UInt16' and 'UInt32'. Please note the download utility is case sensitive and will not accept e.g. 'uint8' or 'uInt16' as valid data types.
eeeeeeeeee	is the actual value of the variable stored at aaaa,bb. The value is either in decimal or in hexadecimal. In the latter case, the string '0x' must precede the value.

NOTE: Each field must be separated by at least one white space. The download utility is case sensitive, the keywords 'Var' and 'ReadWrite' must be entered as shown.

Examples of valid entries:

Dictionary Index	Sub-Index	Description	Key-word	Type	Access	Data
1401	0	number of entries	Var	UInt8	ReadWrite	0x2
1401	1	COB-ID	Var	UInt32	ReadWrite	0x206
1401	2	Transmission type	Var	UInt8	ReadWrite	255
1604	1	PDO Mapping 1(Rx)	Var	UInt32	ReadWrite	0x31000010
100C		GuardTime	Var	UInt16	ReadWrite	0x64

Table A.1 in Appendix shows the complete Object Dictionary supported by the HE693CAN501 module.

IMPORTANT REMARK:

For correct processing of the 'RPT' configuration file by the LoadMod programming utility, the dictionary objects must be listed in ascending order as far as their addresses are concerned (Index, Sub-Index). Otherwise the HE-693CAN501 module will behave in an unexpected way.
This rule must be strictly followed for the data of the Master device. For the slave devices, you must ensure that all objects up to the address 0x100e are in numerical order, beyond the address 0x100e you are free to list your configuration objects in any order.