AVTRON
RS-485 (MULTIPROTOCOL)
OPTION BOARD

(For Use with ACCel500 Frequency Converters)

August 23, 2006

AVTRON MANUFACTURING, INC.
Cleveland, Ohio

AVTRON
RS-485 (MULTIPROTOCOL) OPTION BOARD

TABLE OF CONTENTS

AVTRON
RS-485 (MULTIPROTOCOL) OPTION BOARD

SECTION I

GENERAL INFORMATION

Instead of sending and receiving information to and from frequency converters through I/O, you can connect them to a fieldbus.

Avtron ACCel500 frequency converters can be connected to the RS-485 bus using a fieldbus board. The converter can then be controlled, monitored and programmed from the host system.

If you purchase your RS-485 Option Board separately, please note that it can be installed only in slot E on the control board of the frequency converter.

****

WARNING

Internal components and circuit boards are at high potential when the frequency converter is connected to the power source. This voltage is extremely dangerous and may cause death or severe injury if you come into contact with it.

****

SECTION II

RS-485 OPTION BOARD TECHNICAL DATA

2-1   GENERAL INFORMATION

TABLE 2-1. RS-485 TECHNICAL DATA

Connections	Interface	**OPTC2**: Pluggable connector (5.08mm) **OPTC8**: 9-pin DSUB connector (female)
	Data transfer method	RS-485, half-duplex
	Transfer cable	Twisted pair (1 pair and shield)
	Electrical isolation	500 VDC
Communications	Modbus RTU Metasys N2	As described in document "Modicon Modbus Protocol Reference Guide."
	Baud rate	300, 600, 1200, 2400, 4800, 9600, 19200 and 38400 kbaud
	Addresses	1 to 247
Environment	Ambient operating temperature	–10°C to 55°C
	Storing temperature	–40°C to 60°C
	Humidity	<95%, no condensation allowed
	Altitude	Max. 1000 m
	Vibration	0.5 G at 9 to 200 Hz
Safety		Fulfills EN50178 standard

<div style="border:1px solid #000; padding:1em; text-align:center">

SECTION III

RS-485 OPTION BOARD
LAYOUT AND CONNECTIONS

</div>

Avtron RS-485 Fieldbus Board is connected to the fieldbus through either a 5-pin pluggable bus connector (board OPTC2) or a 9-pin female sub-D-connector (board OPTC8). The communication with the control board of the frequency converter takes place through the standard Avtron Interface Board Connector.

3-1   RS-485 OPTC2 OPTION BOARD
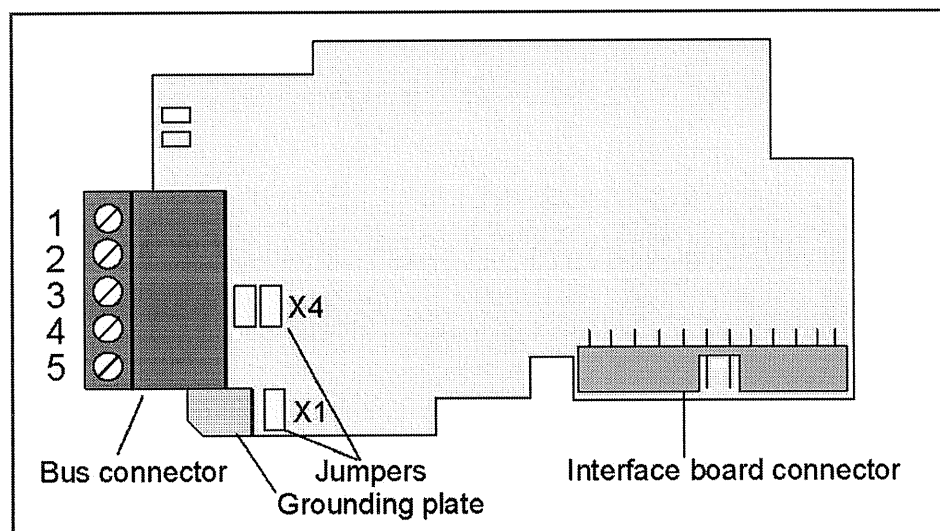
Figure 3-1. RS-485 Option Board OPTC2

TABLE 3-1. OPTC2 BUS CONNECTOR SIGNALS

Signal	Connector	Description
NC*	1*	No connection
VP	2	Supply voltage – plus (5V)
RxD/TxD –N	3	Receive/Transmit data – A
RxD/TxD –P	4	Receive/Transmit data – B
DGND	5	Data ground (reference potential for VP)
*You can use this pin (1) to bypass the cable shield to the next slave		

3-1

3-2   RS-485 OPTC8 OPTION BOARD
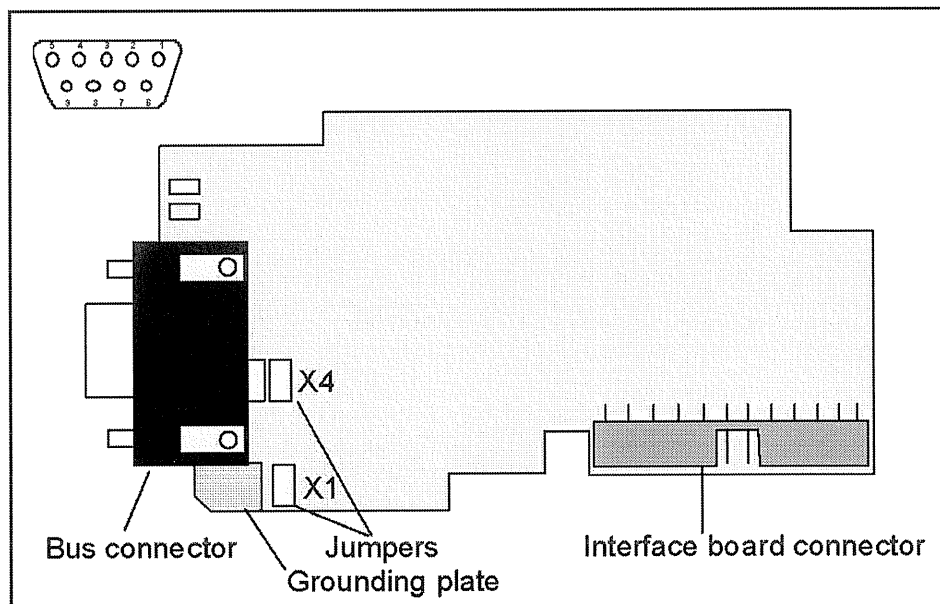
Figure 3-2. RS-485 Option Board OPTC8

TABLE 3-2. OPTC8 BUS CONNECTOR SIGNALS

Signal	Connector	Description
Shield	1	Cable shield
RxD/TxD-N	3	Receive/ A
DGND	5	Data ground (reference potential for VP)
VP	6	Supply voltage – plus (5V)
RxD/TxD-P	8	Receive/ Transmit data/ B

3-3   GROUNDING

3-3.1   GROUNDING BY CLAMPING THE CABLE TO THE CONVERTER FRAME

This manner of grounding is the most effective and especially recommended when the distances between the devices are relatively short or if the device is the last device on the net. Normally, the option board has already been installed in slot D or slot E of the control board. It is not necessary to detach the whole board for the grounding of the bus cable shield. Just detach the terminal block.

1. Strip about 5 cm of the cable and cut off the grey cable shield. Remember to do this for both bus cables (except for the last device). See Figures 3-3 and 3-4 below.

2. Leave no more than 1 cm of the cable outside the terminal block and strip the data cables at about 0.5 cm to fit in the terminals. See Figure 3-4 below. Note: Do this for both bus cables.
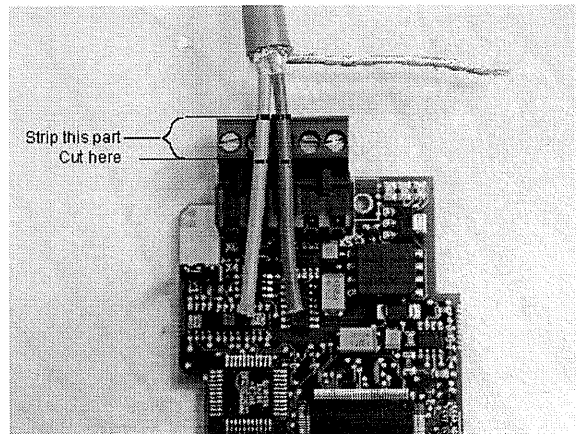
Figure 3-3. Stripping Data Cables
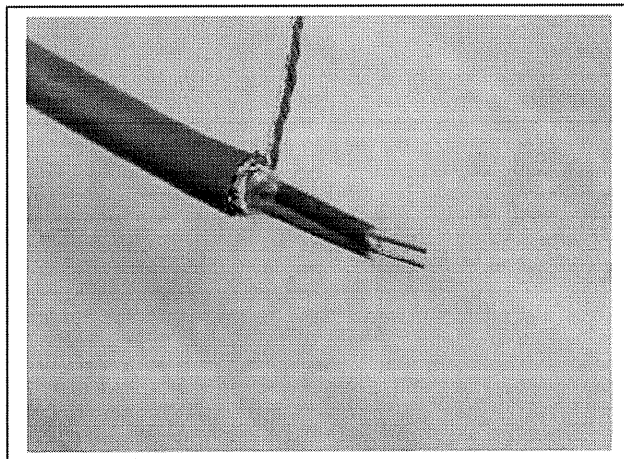
Figure 3-4. Stripping Data Cables

3. Insert the data cables of both cables into terminals #3 (Line B) and #4 (Line A).

4. Strip the cable at such a distance from the terminal that you can fix it to the frame with the grounding clamp. See Figure 3-5.
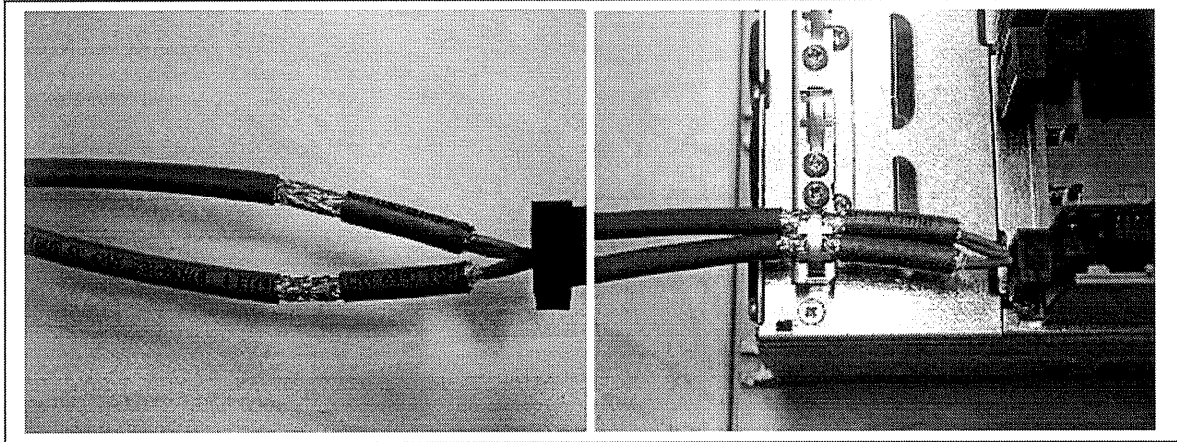
Figure 3-5. Grounding the Data Cables

3-3.2   GROUNDING ONLY ONE POINT ON THE NET

The shield should be connected to ground only at the last device on the net in the same way as described in section 3-3.1. Other devices of the net just pass the shield.

1. Strip about 5 cm of the cable and cut off the grey cable shield. Remember to do this for both bus cables (except for the last device).

2. Leave no more than 1 cm of the cable outside the terminal block and strip the data cables at about 0.5 cm to fit in the terminals. See Figure 3-6. Do this for both bus cables.
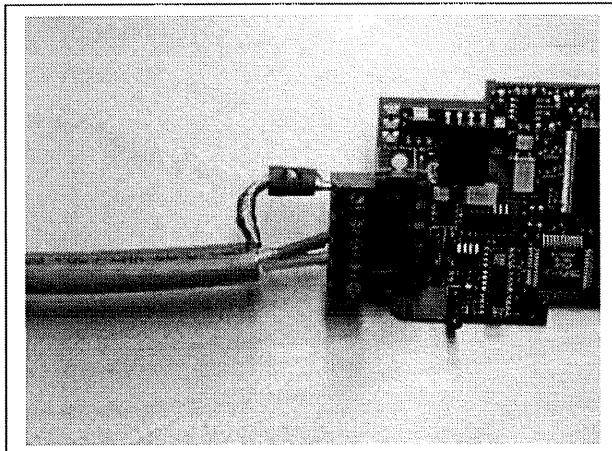
Figure 3-6. Grounding the Last Device on the Net
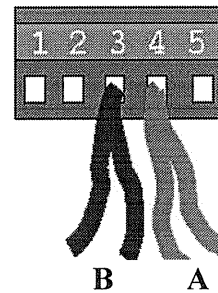
3. Fix both the cables on the frame with the clamp. See Figure 3-7.
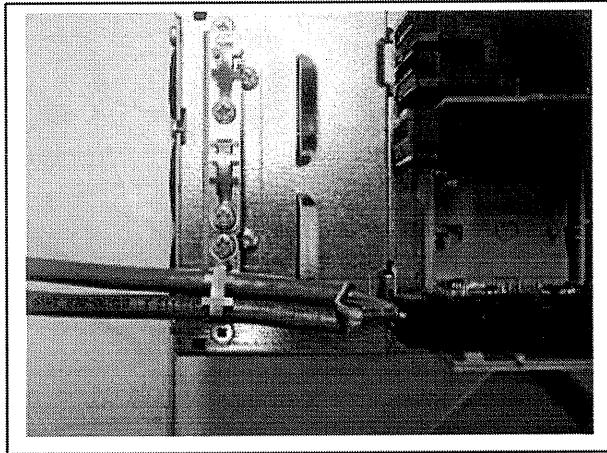
Figure 3-7. Fixing the Cables to the Frame

3-3.3   <u>GROUNDING JUMPER X1</u>

The Grounding jumper X1 on the OPTC8 is used for grounding selection. If position ON is selected, it means that the D-sub connector PIN1 is connected directly to ground. If position OFF is selected, it means that PIN1 is connected to ground via an RC-filter. Jumper X1 has no effect on OPTC2.
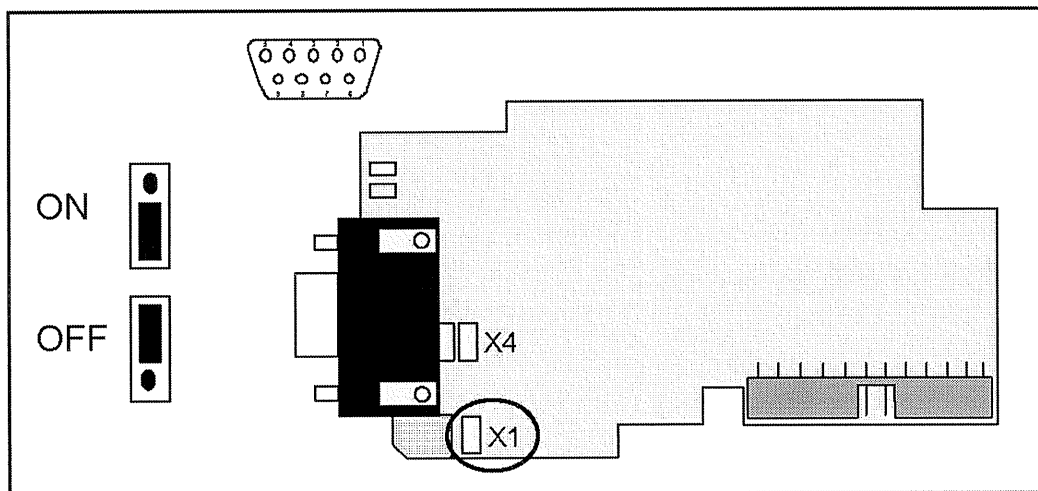
Figure 3-8. Grounding Jumper X1

3-4   BUS TERMINAL RESISTORS

If the RS-485 Option Board is the last device of the fieldbus line, the bus termination must be set. Use jumper X4 (ON position) or external termination resistors (e.g. in DSUB-9 connector). See Figure 3-9.
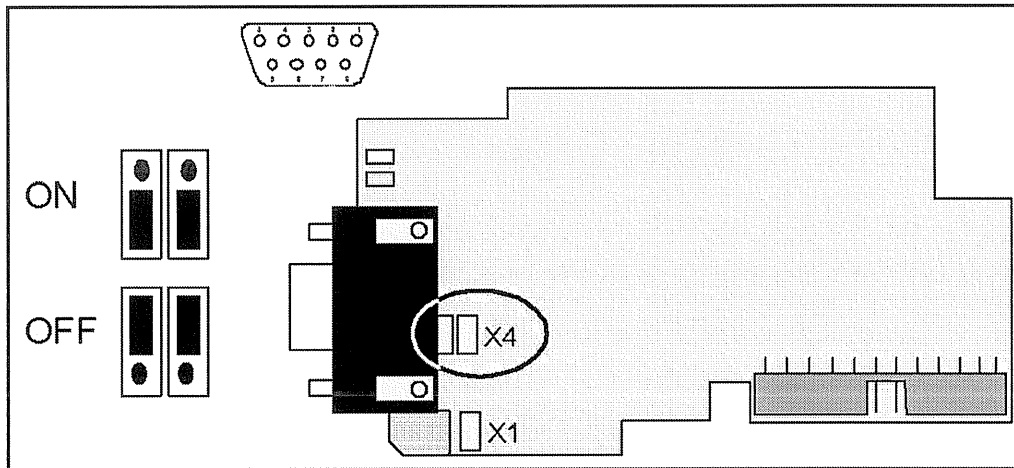
Figure 3-9. Using Jumper X4 to Set the Bus Termination

3-5   BUS BIASING

Bus biasing is required to ensure faultless communication between devices at RS-485 bus. Bus biasing makes sure that the bus state is at proper potential when no device is transmitting. Without biasing, faulty messages can be detected when the bus is in idle state. RS-485 bus state should be neither +0.200 to +7 V or −0.200 to -7 V. Illegal bus state is <200 mV to -200 mV.

TABLE 3-3. BIAS RESISTOR SIZE vs
NUMBER OF NODES

Number of Nodes	Bias Resistance
2-5	1.8 kOhm
5-10	2.7 kOhm
11-20	12 kOhm
21-30	18 kOhm
31-40	27 kOhm

Fail Safe Biasing in OPTC2 Option Board

Connect biasing resistors between pins #2 and #4 as well as pins #3 and #5 as shown in Figure 3-10.
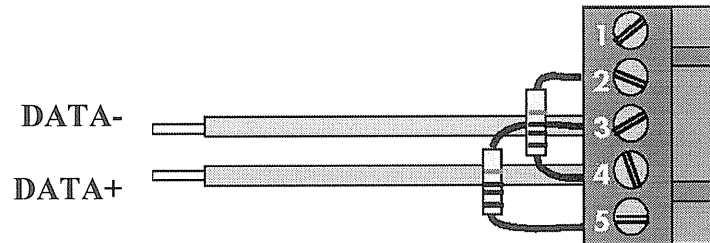
Figure 3-10. Connecting Biasing Resistors

Matters related to this are discussed in the application note Failsafe Biasing of Differential Buses (an-847.pdf) published by National Semiconductor.

3-6   LED INDICATIONS

The two LED indications next to the connector show the present statuses of the RS-485 board (yellow) and the Fieldbus Module (green).
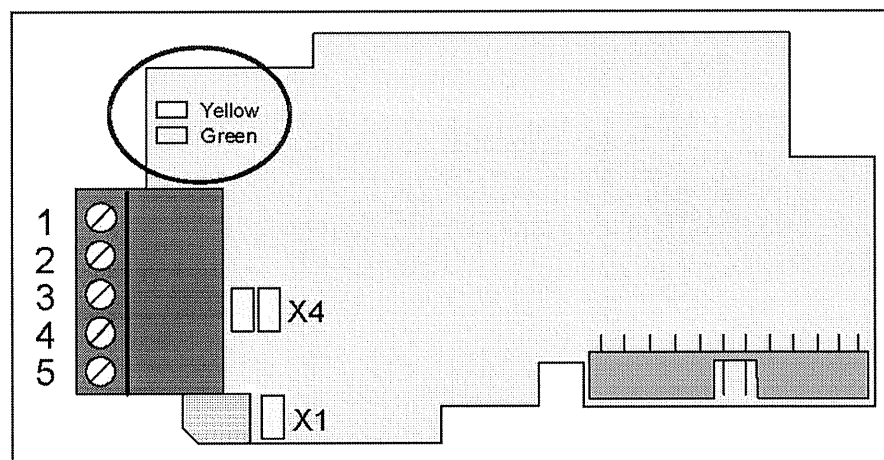
Figure 3-11. LED Indications on the RS-485 Board

RS-485 Board Status LED (BS) YELLOW

LED is:	Meaning:
OFF	Option board not activated
ON	Option board in initialization state waiting for activation command from the frequency converter
Blinking fast (once/sec)	Option board is activated and in RUN state • Option board is ready for external communication
Blinking slow (once/5 secs)	Option board is activated and in FAULT state • Internal fault of option board

Fieldbus Status LED (FS) GREEN

LED is:	Meaning:
OFF	Fieldbus module is waiting for parameters from the frequency converter • No external communication
ON	Fieldbus module is activated • Parameters received and module activated • Module is waiting for messages from the bus
Blinking fast (once/sec)	Module is activated and receiving messages from the bus
Blinking slow (once/5 secs)	Module is in FAULT state • No messages from Master within the watchdog time • Bus broken, cable loose or Master off line

SECTION IV

INSTALLATION OF RS-485 OPTION BOARD

A	ACCel500 frequency converter.	
B	Remove the cable cover.	

C	Open the cover of the control unit.	
D	Install RS-485 option board in slot E on the control board of the frequency converter. Make sure that the grounding plate (see below) fits tightly in the clamp.	
E	Make a sufficiently wide opening for your cable by cutting the grid as wide as necessary.	
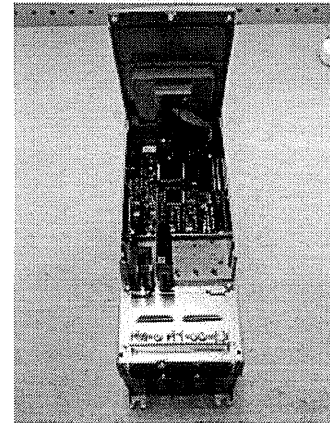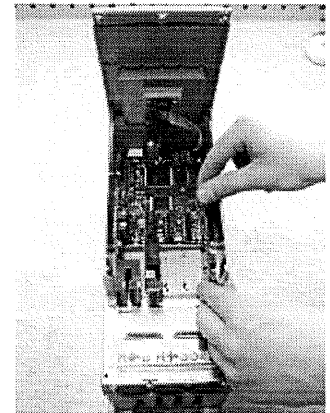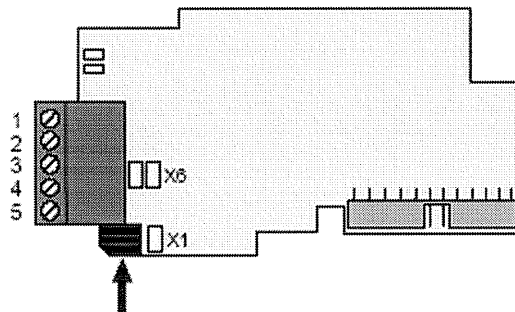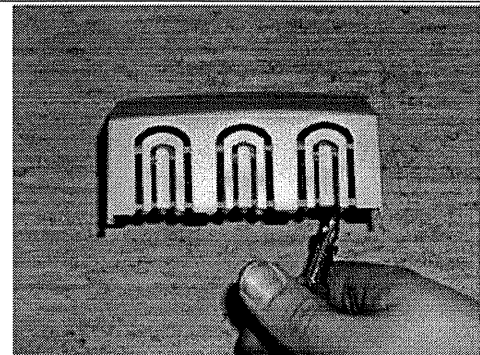
| F | Close the cover of the control unit and the cable cover. | |

SECTION V
COMMISSIONING

Before beginning, read Section VII, Commissioning, in the ACCel500 Frequency Converters manual.

NOTE

You must select Fieldbus as the active control place, if you wish to control the frequency converter through fieldbus. See the ACCel500 Software manual.

5-1   FIELDBUS BOARD PARAMETERS

The Avtron RS-485 board is commissioned with the control keypad by giving values to appropriate parameters in menu M7 (for locating the expander board menu, see ACCel500 Software manual).

Expander Board Menu (M7)

The Expander board menu makes it possible for the user 1) to see what expander boards are connected to the control board and 2) to reach and edit the parameters associated with the expander board.

Enter the following menu level (G#) with the Menu button right. At this level, you can browse through slots A to E with the Browser buttons to see what expander boards are connected. On the lowermost line of the display, you also see the number of parameter groups associated with the board.

If you press the Menu button right once more, you will reach the parameter group level where there are two groups: Editable parameters and Monitored values. A further press on the Menu button right takes you to either of these groups.

RS-485 Parameters

To commission the RS-485 board, enter the level P7.5.1.# from the Parameters group (G7.5.1).
Give desired values to all RS-485 parameters (see Figure 5-1 and Table 5-1).
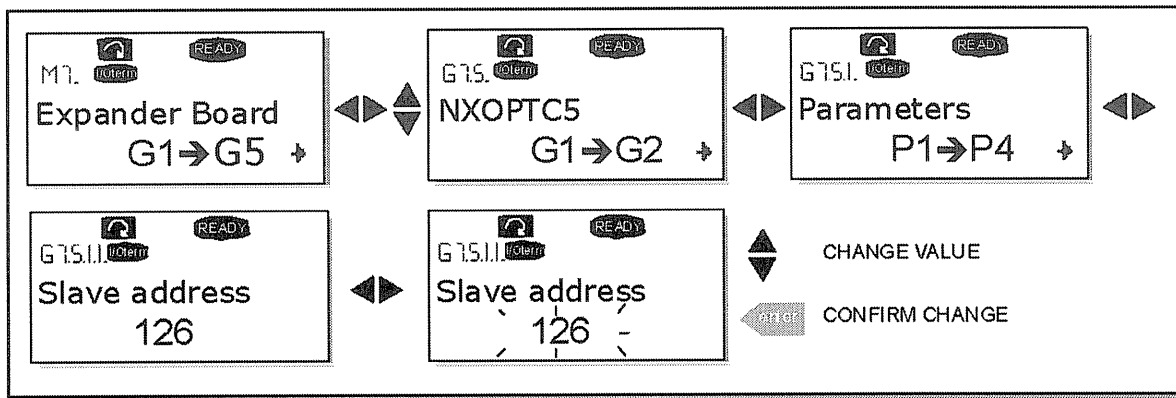
Figure 5-1. Changing the RS-485 Board Commissioning Parameter Values

TABLE 5-1. RS-485 PARAMETERS

#	Name	Default	Range	Description
1	COMMUNICATION PROTOCOL	1	1 – Modbus RTU 2 – N2	Protocol
2	SLAVE ADDRESS	1	1 to 247	
3	BAUD RATE	6	1 – 300 baud 2 – 600 baud 3 – 1200 baud 4 – 2400 baud 5 – 4800 baud 6 – 9600 baud 7 – 19200 baud 8 – 38400 baud	Communication speed When N2 protocol is used Baudrate must be set to 9600.
4	PARITY TYPE	0	0 – None 1 – Even 2 – Odd	Describes what kind of parity checking is used. When N2-protocol is used, Parity type must be set to 0 = None
5	COMMUNICATION TIMEOUT	20	0—OFF 1—300 s	See Communication Timeout paragraph below.
6	OPERATE MODE	1	1 – Normal	Reserved for later use

The parameters of every device must be set before connecting to the bus. Especially the
parameters Communication Protocol, Slave Address and Baud Rate must be the same as in the
master configuration.

Communication Timeout

The RS-485 board initiates a communication error if communication is broken for as long as defined by the Communication Timeout. Communication Timeout is disabled when given the value 0.

Communication Status

To see the present status of the RS-485 fieldbus, enter the Comm.Status page from Monitor menu (G7.5.2). See Figure 5-2 and Table 5-2 below.
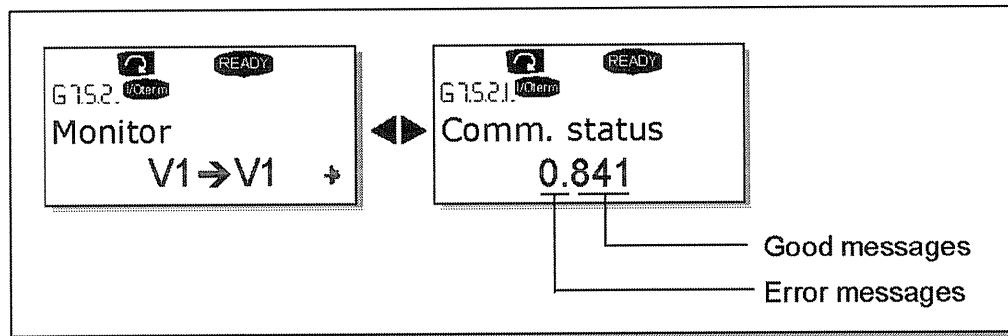
Figure 5-2. Communication Status

TABLE 5-2. RS-485 MESSAGE INDICATIONS

Good messages	
0 to 999	Number of messages received without communication errors
Error messages	
0 to64	Number of messages received with CRC or parity errors

SECTION VI

MODBUS PROTOCOL

6-1   INTRODUCTION TO MODBUS RTU PROTOCOL

The Modbus protocol is an industrial communications and distributed control system to integrate PLCs, computers, terminals, and other monitoring, sensing, and control devices. Modbus is a Master-Slave communications protocol. The Master controls all serial activity by selectively polling one or more slave devices. The protocol provides for one master device and up to 247 slave devices on a common line. Each device is assigned an address to distinguish it from all other connected devices.

The Modbus protocol uses the master-slave technique, in which only one device (the master) can initiate a transaction. The other devices (the slaves) respond by supplying the request data to the master, or by taking the action requested in the query. The master can address individual slaves or initiate a broadcast message to all slaves. Slaves return a message ('response') to queries that are addressed to them individually. Responses are not returned to broadcast queries from the master.

A transaction comprises a single query and single response frame or a single broadcast frame. The transaction frames are defined below.
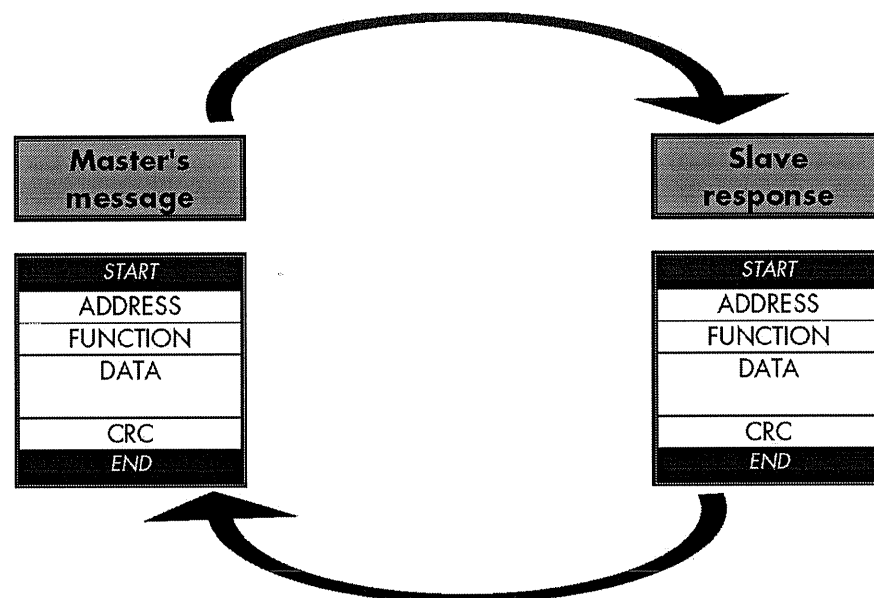
Figure 6-1. The Basic Structure of a Modbus Frame

Valid slave device addresses are in the range of 0 to 247 decimal. The individual slave devices are assigned addresses in the range of 1 to 247. A master addresses a slave by placing the slave address in the address field of the message. When the slave sends its response, it places its own address in this address field of the response to let the master know which slave is responding.

The function code field of a message frame contains two characters (ASCII) or eight bits (RTU). Valid codes are in the range of 1 to 255 decimal. When a message is sent from a master to a slave device the function code field tells the slave what kind of action to perform. Examples are to read the ON / OFF states of a group of discrete coils or inputs; to read the data contents of a group of registers; to read the diagnostic status of the slave; to write to designated coils or registers; or to allow loading, recording, or verifying the program within the slave.

When the slave responds to the master, it uses the function code field to indicate either a normal (error-free) response or that some kind of error occurred (called an exception response). For a normal response, the slave simply echoes the original function code. For an exception response, the slave returns a code that is equivalent to the original function code with its most significant bit set to a logic 1.

The data field is constructed using sets of two hexadecimal digits, in the range of 00 to FF hexadecimal. These can be made from a pair of ASCII characters, or from one RTU character, according to the network's serial transmission mode.

The data field of messages sent from a master to slave devices contains additional information which the slave must use to take the action defined by the function code. This can include items like discrete and register addresses, the quantity of items to be handled, and the count of actual data bytes in the field. If no error occurs, the data field of a response from a slave to a master contains the data requested. If an error occurs, the field contains an exception code that the master application can use to determine the next action to be taken.

Two kinds of checksum are used for standard Modbus networks. The error checking field contents depend upon the transmission method that is being used.

6-1.1   SUPPORTED FUNCTIONS

TABLE 6-1. SUPPORTED MESSAGES

Function Code	Description	Address range
03	Read Holding Registers	Applies to all addresses
04	Read Input Registers	Applies to all addresses
06	Write Single Register	Applies to all addresses
16	Write Multiple Registers	Applies to all addresses
Note: Broadcasting can be used with codes 06 and 16		

6-1.1.1   Read Holding Registers

The query message specifies the starting register and the quantity of registers to be read. Registers are addressed starting with zero, i.e. registers 1 to 16 are addressed as 0 to 15.

Example of a request to read registers 42001-42003 from Slave device 1:

ADDRESS		01 hex	Slave address 1 hex (= 1)
FUNCTION		03 hex	Function 03 hex (= 3)
DATA	**Starting address HI**	07 hex	Starting address 07d0 hex (= 2000)
	Starting address LO	D0 hex	
	No. of points HI	00 hex	Number of registers 0003 hex (= 3)
	No. of points LO	03 hex	
ERROR	**CRC HI**	05 hex	CRC field 0546 hex (= 1350)
CHECK	**CRC LO**	46 hex	

6-1.1.2   Read Input Registers

The query message specifies the starting register and the quantity of registers to be read. Registers are addressed starting with zero, i.e. registers 1 to 16 are addressed as 0 to 15.

Example of a request to read registers 32001 from Slave device 1:

ADDRESS		01 hex	Slave address 1 hex (= 1)
FUNCTION		04 hex	Function 04 hex (= 4)
DATA	**Starting address HI**	07 hex	Starting address 07d0 hex (= 2000)
	Starting address LO	D0 hex	
	No. of points HI	00 hex	Number of registers 0003 hex (= 3)
	No. of points LO	01 hex	
ERROR	**CRC HI**	31 hex	CRC field 3147 hex (= 12615)
CHECK	**CRC LO**	47 hex	

6-1.1.3   Preset Single Register

The query message specifies the register reference to be preset. Registers are addressed starting with zero, i.e. register 1 is addressed as 0.

Example of a request to preset register 42001 to 00001hex in Slave device 1:

ADDRESS		01 hex	Slave address 1 hex (= 1)
FUNCTION		06 hex	Function 04 hex (= 4)
DATA	Starting address HI	07 hex	Starting address 07d0 hex (= 2000)
	Starting address LO	D0 hex	
	No. of points HI	00 hex	Data = 0001 hex (= 1)
	No. of points LO	01 hex	
ERROR	CRC HI	48 hex	CRC field 4887 hex (= 18567)
CHECK	CRC LO	87 hex	

6-1.1.4   Preset Multiple Registers

The query message specifies the register references to be preset. Registers are addressed starting with zero, i.e. register 1 is addressed as 0.

Example of a request to preset two registers starting at 42001 to 0001hex and 0010hex in Slave device 1:

ADDRESS		01 hex	Slave address 1 hex (= 1)
FUNCTION		10 hex	Function 10 hex (= 16)
DATA	Starting address HI	07 hex	Starting address 07d0 hex (= 2000)
	Starting address LO	D0 hex	
	No. of points HI	00 hex	Number of registers 0002 hex (= 2)
	No. of points LO	02 hex	
	Byte Count	04 hex	Byte count 04 hex (= 4)
	Data HI	00 hex	Data 1 = 0001 hex (= 1)
	Data LO	01 hex	
	Data HI	00 hex	Data 2 = 0010 hex (= 16)
	Data LO	10 hex	
ERROR	CRC HI	88 hex	CRC field 88CF hex (= 35023)
CHECK	CRC LO	CF hex	

6-1.2   EXCEPTION RESPONSES

Error response is given when the Slave receives a message without communication errors, but cannot handle it. Examples of such messages are an incorrect register address, data value or unsupported message. No answer is given if a CRC or parity error occurs or the message is a broadcast message.

TABLE 6-2. EXCEPTION RESPONSE CODES

Code	Function	Description
01	ILLEGAL FUNCTION	The message function requested is not recognized by the slave.
02	ILLEGAL DATA ADDRESS	The received data address is not an allowable address for the slave.
03	ILLEGAL DATA VALUE	The received data value is not an allowable value for the slave.
06	SLAVE DEVICE BUSY	The message was received without error but the slave was engaged in processing a long duration program command.

Example of an Exception Response

In an exception response, the Slave sets the most-significant bit (MSB) of the function code to 1. The Slave returns an exception code in the data field.

Command Master – Slave:

ADDRESS		01 hex	Slave address 1 hex (= 1)
FUNCTION		04 hex	Function 4 hex (= 4)
DATA	**Starting address HI**	17 hex	Starting address 1770 hex (= 6000)
	Starting address LO	70 hex	
	No. of registers HI	00 hex	Invalid number of registers 0005 hex (= 5)
	No. of registers LO	05 hex	
ERROR	**CRC HI**	34 hex	CRC field 3466 hex (= 13414)
CHECK	**CRC LO**	66 hex	

Message Frame:

01	04	17	70	00	05	34	66

Answer Slave – Master:

ADDRESS		01 hex	Slave address 1 hex (= 1)
FUNCTION		14 hex	Most significant bit set to 1
ERROR CODE		02 hex	Error code 02 >= Illegal Data Address
ERROR	**CRC HI**	AE hex	CRC field AEC1 hex (= 44737)
CHECK	**CRC LO**	C1 hex	

Reply Frame:

01	14	02	AE	C1

6-2   MODBUS INTERFACE

Features of the Modbus-Avtron interface:

- Direct control of ACCel500 frequency converter (e.g. Run, Stop, Direction, Speed reference, Fault reset)
- Full access to all ACCel500 Frequency converter parameters
- Monitor ACCel500 status (e.g. Output frequency, Output current, Fault code)

6-2.1   MODBUS REGISTERS

The Avtron variables and fault codes as well as the parameters can be read and written from Modbus. The parameter addresses are determined in the application. Every parameter and actual value has been given an ID number in the application. The ID numbering of the parameter as well as the parameter ranges and steps can be found in the application manual in question. The parameter value shall be given without decimals. If several parameters/actual values are read with one message, the addresses of the parameters/actual values must be consecutive.

All values can be read with function codes 3 and 4 (all registers are 3X and 4X reference). Modbus registers are mapped to drive IDs as follows:

TABLE 6-3. INDEX TABLE

ID	Modbus register	Group	R/W
1 to 98	40001 to 40098 (30001 to 30098)	Actual Values	30/1
99	40099 (30099)	Fault Code	30/1
101 to 1999	40101 to 41999 (30101 to 31999)	Parameters	30/1
2001 to 2099	42001 to 42099 (32001 to 32099)	Process Data In	20/20
2101 to 2199	42101 to42199 (32101 to 32199)	Process Data Out	20/20

6-2.2   PROCESS DATA

The process data fields are used to control the drive (e.g. Run, Stop , Reference, Fault Reset) and to quickly read actual values (e.g. Output frequency, Output current, Fault code). The fields are structured as follows:

TABLE 6-4. PROCESS DATA SLAVE → MASTER (max 22 bytes)

ID	Modbus Register	Name	Range/Type
2101	32101, 42101	FB Status Word	Binary coded
2102	32102, 42102	FB General Status Word	Binary coded
2103	32103, 42103	FB Actual Speed	0 to 10000 %
2104	32104, 42104	FB Process Data Out 1	See appendix 1
2105	32105, 42105	FB Process Data Out 2	See appendix 1
2106	32106, 42106	FB Process Data Out 3	See appendix 1
2107	32107, 42107	FB Process Data Out 4	See appendix 1
2108	32108, 42108	FB Process Data Out 5	See appendix 1
2109	32109, 42109	FB Process Data Out 6	See appendix 1
2110	32110, 42110	FB Process Data Out 7	See appendix 1
2111	32111, 42111	FB Process Data Out 8	See appendix 1

TABLE 6-5. PROCESS DATA MASTER → SLAVE (max 22 bytes)

ID	Modbus Register	Name	Range/Type
2001	32001, 42001	FB Control Word	Binary coded
2002	32002, 42002	FB General Control Word	Binary coded
2003	32003, 42003	FB Speed Reference	0 to 10000 %
2004	32004, 42004	FB Process Data In 1	Integer 16
2005	32005, 42005	FB Process Data In 2	Integer 16
2006	32006, 42006	FB Process Data In 3	Integer 16
2007	32007, 42007	FB Process Data In 4	Integer 16
2008	32008, 42008	FB Process Data In 5	Integer 16
2009	32009, 42009	FB Process Data In 6	Integer 16
2010	32010, 42010	FB Process Data In 7	Integer 16
2011	32011, 42011	FB Process Data In 8	Integer 16

The use of process data depends on the application. In a typical situation, the device is started and stopped with the ControlWord (CW) written by the Master and the Rotating speed is set with Reference (REF). With PD1 to PD8 the device can be given other reference values (e.g., Torque reference). With the StatusWord (SW) read by the Master, the status of the device can be seen. Actual Value (ACT) and PD1 to PD8 show the other actual values.

6-2.3   PROCESS DATA IN

This register range is reserved for the control of the frequency converter. Process data in is located in range ID 2001 to 2099. The registers are updated every 10 ms. See Table 6-6.

TABLE 6-6. FIELDBUS BASIC INPUTS

ID	Modbus register	Name	Range/Type
2001	32001, 42001	FB Control Word	Binary coded
2002	32002, 42002	FB General Control Word	Binary coded
2003	32003, 42003	FB Speed Reference	0…10000 %
2004	32004, 42004	FB Process Data In 1	Integer 16
2005	32005, 42005	FB Process Data In 2	Integer 16
2006	32006, 42006	FB Process Data In 3	Integer 16
2007	32007, 42007	FB Process Data In 4	Integer 16
2008	32008, 42008	FB Process Data In 5	Integer 16
2009	32009, 42009	FB Process Data In 6	Integer 16
2010	32010, 42010	FB Process Data In 7	Integer 16
2011	32011, 42011	FB Process Data In 8	Integer 16

6-2.3.1   Control Word

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
–	–	–	–	–	–	–	–	–	–	–	–	–	RST	DIR	RUN

In Avtron applications, the three first bits of the control word are used to control the frequency converter. However, you can customize the content of the control word for your own applications because the control word is sent to the frequency converter as such.

TABLE 6-7. CONTROL WORD BIT DESCRIPTIONS

Bit	Description	
	Value = 0	Value = 0
0	Stop	Run
1	Clockwise	Counterclockwise
2	Rising edge of this bit will reset active fault	
3 to 15	Not in use	Not in use

6-2.3.2   Speed Reference

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
MSB															LSB

This is the Reference 1 to the frequency converter. Used normally as Speed reference. The allowed scaling is –10000 to 10000. In the application, the value is scaled in percentage of the frequency area between the set minimum and maximum frequencies.

6-2.3.3   Process Data In 1 to 8

Process Data In values 1 to 8 can be used in applications for various purposes. Update rate is 10 ms for all values. See ACCel500 Adjustable Frequency Drives Application manual for usage of these data values.

6-2.4   PROCESS DATA OUT

This register range is normally used to fast monitoring of the frequency converter. Process data out is located in range ID 2101 to 2199.

TABLE 6-8. FIELDBUS BASIC OUTPUT TABLE

ID	Modbus register	Name	Range/Type
2101	32101, 42101	FB Status Word	Binary coded
2102	32102, 42102	FB General Status Word	Binary coded
2103	32103, 42103	FB Actual Speed	0 to 10000 %
2104	32104, 42104	FB Process Data Out1	See appendix 1
2105	32105, 42105	FB Process Data Out2	See appendix 1
2106	32106, 42106	FB Process Data Out3	See appendix 1
2107	32107, 42107	FB Process Data Out4	See appendix 1
2108	32108, 42108	FB Process Data Out5	See appendix 1
2109	32109, 42109	FB Process Data Out6	See appendix 1
2110	32110, 42110	FB Process Data Out7	See appendix 1
2111	32111, 42111	FB Process Data Out8	See appendix 1

6-2.4.1   Status Word

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
–	–	–	–	–	UVFS	DDIR	TCSPDL	FR	Z	AREF	W	FLT	DIR	RUN	RDY

Information about the status of the device and messages is indicated in the Status word. The Status word is composed of 16 bits that have the following meanings:

TABLE 6-9. STATUS WORD BIT DESCRIPTIONS

Bit	Description	
	Value = 0	Value = 0
0	Not Ready	Ready
1	STOP	RUN
2	Clockwise	Counterclockwise
3	–	Faulted
4	–	Warning
5	Ref. frequency not reached	Ref. frequency reached
6	–	Motor is running at zero speed
7	Flux Ready	Flux Not Ready
8	TC Speed Limit Active	TC Speed Limit Not Active
9	Detected Encoder Direction Clockwise	Encoder Direction Counterclockwise
10	UV Fast Stop Active	UV Fast Stop Not Active
11 to 15	Not in use	Not in use

6-2.4.2   General Status Word

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
I/O	PANEL	FB	–	–	–	–	–	–	–	–	–	–	–	–	–

TABLE 6-10. GENERAL STATUS WORD BIT DESCRIPTIONS

Bit	Description
0 to 12	Not in use
13	Fieldbus control, (1 = FB control active)
14	Panel control, (1 = Panel control active)
15	I/O Control, (1 = I/O control active)

6-2.4.3   Actual Speed

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
MSB															LSB

This is the reference 1 to the frequency converter. Used normally as Speed reference. The allowed scaling is –10000 to 10000. In the application, the value is scaled in percentage of the frequency area between set minimum and maximum frequency.

6-2.4.4   Process Data Out 1 to 8

Process Data Out values 1 to 8 can be used in application for various purposes. Update rate is 10ms for all values. See Appendix A for use of these values.

6-2.5 PARAMETERS

The parameter addresses are determined in the application. Every parameter has been given an ID number in the application. The ID numbering of the parameter as well as the parameter ranges and steps can be found in the application manual in question. The parameter value shall be given without decimals. The following functions can be activated with parameters:

TABLE 6-11. PARAMETERS

Function code	Function	Modbus Address	Parameter ID's
03	Read Holding Registers	30101 to 31999	101-1999
04	Read Input Registers	40101 to 41999	101-1999
06	Preset Single Register	40101 to 41999	101-1999
16	Preset Multiple Registers	40101 to 41999	101-1999

6-2.6 ACTUAL VALUES

The actual values as well as parameter addresses are determined in the application. Every actual value has been given an ID number in the application. The ID numbering of the actual values as well as the value ranges and steps can be found in the application manual in question. The following functions can be activated with parameters:

TABLE 6-12. ACTUAL VALUES

Function code	Function	Actual values
03	Read Holding Registers	30001-30098
04	Read Input Registers	40001-40098

6-11

6-2.7   EXAMPLE MESSAGES

Example 1

Write the process data 42001 to 42003 with command 16 (Preset Multiple Registers).

Command Master – Slave:

ADDRESS		01 hex	Slave address 1 hex (= 1)
FUNCTION		10 hex	Function 10 hex (= 16)
DATA	**Starting address HI**	07 hex	Starting address 07d0 hex (= 2000)
	Starting address LO	D0 hex	
	No. of registers HI	00 hex	Number of registers 0003 hex (= 3)
	No. of registers LO	03 hex	
	Byte Count	06 hex	Byte count 06 hex (= 6)
	Data HI	00 hex	Data 1 = 0001 hex (= 1). Setting control word run bit to 1.
	Data LO	01 hex	
	Data HI	00 hex	Data 2 = 0000 hex (= 0). General control word 0.
	Data LO	00 hex	
	Data HI	13 hex	Data 3 = 1388 hex (= 5000), Speed Reference to 50.00%
	Data LO	88 hex	
ERROR	**CRC HI**	C8 hex	CRC field C8CB hex (= 51403)
CHECK	**CRC LO**	CB- hex	

Message Frame:

01	10	07	D0	00	03	06	00	01	00	00	13	88	C8	CB

The reply to Preset Multiple Registers message is the echo of 6 first bytes.

Answer Slave – Master:

ADDRESS		01 hex	Slave address 1 hex (= 1)
FUNCTION		10 hex	Function 10 hex (= 16)
DATA	**Starting address HI**	07 hex	Starting address 07d0 hex (= 2000)
	Starting address LO	D0 hex	
	No. of registers HI	00 hex	Number of registers 0003 hex (= 3)
	No. of registers LO	03 hex	
ERROR	**CRC HI**	F1 hex	CRC F101 hex (= 61697)
CHECK	**CRC LO**	01 hex	

Message Frame:

01	10	07	D0	00	03	F1	01

Example 2

Read the Process Data 42103 to 42104 with command 4 (Read Input Registers).

Command Master – Slave:

ADDRESS		01 hex	Slave address 1 hex (= 1)
FUNCTION		04 hex	Function 4 hex (= 4)
DATA	**Starting address HI**	08 hex	Starting address 0836 hex (= 2102)
	Starting address LO	36 hex	
	No. of registers HI	00 hex	Number of registers 0002 hex (= 2)
	No. of registers LO	02 hex	
ERROR	**CRC HI**	93 hex	
CHECK	**CRC LO**	A5 hex	CRC field B321 hex (= 45857)

Message Frame:

01	04	08	36	00	02	93	A5

The reply to the Read Input Registers message contains the values of the read registers.

Answer Slave – Master:

ADDRESS		01 hex	Slave address 1 hex (= 1)
FUNCTION		04 hex	Function 4 hex (= 4)
DATA	**Byte Count**	02 hex	Byte count 4 hex (= 4)
	Data HI	13 hex	Speed reference = 1388 hex (=5000 => 50.00%)
	Data LO	88 hex	
	Data HI	09 hex	Output Frequency = 09C4 hex (=2500 =>25.00Hz)
	Data LO	C4 hex	
ERROR	**CRC HI**	F0 hex	CRC field B321 hex (= 45857)
CHECK	**CRC LO**	E9 hex	

Message Frame:

01	04	02	13	88	09	C4	F0	E9

6-3   START-UP TEST

Frequency converter application

Choose Fieldbus (Bus/Comm) as the active control place (see the ACCel500 Software manual).

Master Software

1. Set FB Control Word (MBaddr 42001) value to 1hex.

2. Frequency converter status is RUN.

3. Set FB Speed Reference (MBaddr 42003) value to 5000 (=50,00%).

4. The Actual value is 5000 and the frequency converter output frequency is 25,00 Hz.

5. Set FB Control Word (MBaddr 42001) value to 0hex.

6. Frequency converter status is STOP.

If FB Status Word (Addr 42101) bit 3 = 1, Status of frequency converter is FAULT.

<div style="border: 1px solid black; padding: 20px;">

SECTION VII

METASYS N2 PROTOCOL

</div>

7-1   INTRODUCTION TO METASYS N2 PROTOCOL

The N2 communications protocol is used by Johnson Controls and others to connect terminal unit controllers to supervisory controllers. It is open to any manufacturer and based upon a simple ASCII protocol widely used in the process control industry.

The physical characteristics of the N2 bus are three wire RS-485 with a maximum of 100 devices over a 4,000 foot distance running at 9,600 bps. Logically, the N2 is a master-slave protocol, the supervisory controller normally being the master. Data is partitioned into common HVAC control objects, such as analog input, analog output, binary input and binary output. N2 messaging supports the reading, writing and overriding of these points. Additionally, there are messages defined to perform uploads and downloads of devices as well as direct memory reads and writes.

7-2   METASYS N2 INTERFACE

Features of the N2 Interface:

- Direct control of Drive (e.g. Run, Stop, Direction, Speed reference, Fault reset)
- Full access to necessary parameters
- Monitor Drive status (e.g. Output frequency, Output current, Fault code)
- In standalone operation, or should the polling stop, the overridden values are released after a specified period (about 10 minutes).

7-2.1   ANALOG INPUT (AI)

All Analog Input (AI) points have the following features:

- Support Change of State (COS) reporting based on high and low warning limits.

- Support Change of State (COS) reporting based on high and low alarm limits.

- Support Change of State (COS) reporting based on override status.

- Always considered reliable and never out of range.

- Writing of alarm and warning limit values beyond the range that can be held by the drive's internal variable will result in having that limit replaced by the "Invalid Float" value even though the message is acknowledged. The net result will be the inactivation of the alarm or warning (the same as if the original out of range value was used).

- Overriding is supported from the standpoint that the "Override Active" bit will be set and the value reported to the N2 network will be the overridden value. However, the value in the drive remains unchanged. Therefore, the N2 system should be set up to disallow overriding AI points or have an alarm condition activated when an AI point is overridden.

- Overriding an AI point with a value beyond the limit allowed by the drive's internal variable will result in an "Invalid Data" error response and the override status and value will remain unchanged.

7-2.2   BINARY INPUT (BI)

All Binary Input (BI) points have the following features:

- Support Change of State (COS) reporting based on current state.

- Support Change of State (COS) reporting based on alarm condition.

- Support Change of State (COS) reporting based on override status.

- Always considered reliable.

Overriding is supported from the standpoint that the "Override Active" bit will be set and the value reported to the N2 network will be the overridden value. However, the value in the drive remains unchanged. Therefore, the N2 system should be set up to disallow overriding BI points or have an alarm condition activated when a BI point is overridden.

7-2.3   ANALOG OUTPUT (AO)

All Analog Output (AO) points have the following features:

- Support Change of State (COS) reporting based on override status.

- Always considered reliable.

● Overriding of the AO points is the method used to change a value. Overriding an AO point with a value beyond the limit allowed by the drive's internal variable will result in an "Invalid Data" error response and the override status and value will remain unchanged. If the overridden value is beyond the drive's parameter limit but within the range that will fit in the variable, an acknowledge response is given and the value will be internally clamped to its limit.

● An AO point override copies the override value to the corresponding drive parameter. This is the same as changing the value on the keypad. The value is non-volatile and will remain in effect when the drive is turned off and back on. It also remains at this value when the N2 network "releases" the point. The N2 system always reads the current parameter value.

NOTE

On some N2 systems, the system will not poll the AO point when it is being overridden. In this case, the N2 system will not notice a change in value if the change is made with the keypad. To avoid this, set the point up as a "local control" type and release it once it has been overridden. In this way, the N2 system will monitor the value when not being overridden.

7-2.4   BINARY OUTPUT (BO)

All Binary Output (BO) points have the following features:

● Support Change of State (COS) reporting based on override status.

● Always considered reliable.

● Overriding BO points control the drive. These points are input commands to the drive. When released, the drive's internal value remains at its last overridden value.

7-2.5   INTERNAL INTEGER (ADI)

All Internal Integer (ADI) points have the following features:

● Do not support Change of State (COS) reporting.

● Can be overridden and the "Override Active" bit will be set. However, the Internal value is unchanged (Read Only).

7-3   N2 POINT MAP

TABLE 7-1. ANALOG INPUTS (AI)

NPT	NPA	Description	Units	Note
AI	1	Speed Setpoint	Hz	2 decimals
AI	2	Output Speed	Hz	2 decimals
AI	3	Motor Speed	Rpm	0 decimal
AI	4	Load (power)	%	1 decimal
AI	5	Megawatt Hours	MWh	Total Counter
AI	6	Motor Current	A	2 decimal
AI	7	Bus Voltage	V	0 decimal
AI	8	Motor Volts	V	1 decimal
AI	9	Heatsink Temperature	° C	0 decimal
AI	10	Motor Torque	%	1 decimal
AI	11	Operating Days (trip)	Day	0 decimal
AI	12	Operating Hours (trip)	Hour	0 decimal
AI	13	Kilowatt Hours (trip)	kWh	Trip Counter
AI	14	Torque Reference	%	1 decimal
AI	15	Motor Temperature Rise	%	1 decimal
AI	16	FBProcessDataOut1 [1]	-32768 to +32767	0 decimal
AI	17	FBProcessDataOut2 [1]	-32768 to +32767	0 decimal
AI	18	FBProcessDataOut3 [1]	-32768 to +32767	0 decimal
AI	19	FBProcessDataOut4 [1]	-32768 to +32767	0 decimal
AI	20	FBProcessDataOut5 [1]	-32768 to +32767	0 decimal
AI	21	FBProcessDataOut6 [1]	-32768 to +32767	0 decimal
AI	22	FBProcessDataOut7 [1]	-32768 to +32767	0 decimal
AI	23	FBProcessDataOut8 [1]	-32768 to +32767	0 decimal

[1] These analog inputs are application specific.

TABLE 7-2. BINARY INPUTS (BI)

NPT	NPA	Description	0 =	1 =
BI	1	Ready	Not Ready	Ready
BI	2	Run	Stop	Run
BI	3	Direction	Clockwise	Counterclockwise
BI	4	Faulted	Not Faulted	Faulted
BI	5	Warning	Not Warning	Warning
BI	6	Ref. Frequency reached	False	True
BI	7	Motor running zero speed	False	True
BI	8	General 0 [3]	0	1
BI	9	General 1 [3]	0	1
BI	10	General 2 [3]	0	1
BI	11	General 3 [3]	0	1
BI	12	General 4 [3]	0	1
BI	13	General 5 [3]	0	1
BI	14	General 6 [3]	0	1
BI	15	General 7 [3]	0	1

[3] These binary inputs are application specific. They are read from the drives General Status Word.

TABLE 7-3. ANALOG OUTPUTS (AO)

NPT	NPA	Description	Units	Note
AO	1	Comms Speed	%	2 decimals
AO	2	Current Limit	A	2 decimals
AO	3	Minimum Speed	Hz	2 decimals
AO	4	Maximum Speed	Hz	2 decimals
AO	5	Accel Time	s	1 decimal
AO	6	Decel Time	s	1 decimal
AO	7	FBProcessDataIN 1 [4]	-32768 to +32767	2 decimals
AO	8	FBProcessDataIN 2 [4]	-32768 to +32767	2 decimals
AO	9	FBProcessDataIN 3 [4]	-32768 to +32767	2 decimals
AO	10	FBProcessDataIN 4 [4]	-32768 to +32767	2 decimals

[4] These Analog Outputs are application specific.

TABLE 7-4. BINARY OUTPUTS (BO)

NPT	NPA	Description	0 =	1 =
BO	1	Comms Start/Stop	Stop	Start
BO	2	Comms Forward/Reverse	Forward	Reverse
BO	3	Reset Fault	N/A	Reset
BO	4	FBFixedControlWord Bit_3 [5]	-	-
BO	5	FBFixedControlWord Bit_4 [5]	-	-
BO	6	FBFixedControlWord Bit_5 [5]	-	-
BO	7	FBFixedControlWord Bit_6 [5]	-	-
BO	8	FBFixedControlWord Bit_7 [5]	-	-
BO	9	FBFixedControlWord Bit_8 [5]	-	-
BO	10	FBFixedControlWord Bit_9 [5]	-	-
BO	11	FBFixedControlWord Bit_10 [5]	-	-
BO	12	FBFixedControlWord Bit_11 [5]	-	-
BO	13	FBFixedControlWord Bit_12 [5]	-	-
BO	14	FBFixedControlWord Bit_13 [5]	-	-
BO	15	FBFixedControlWord Bit_14 [5]	-	-
BO	16	FBFixedControlWord Bit_15 [5]	-	-

[5] These Binary Outputs are application specific.

TABLE 7-4. INTERNAL INTEGERS (ADI)

NPT	NPA	Description	Units
ADI	1	Active Fault Code	-

SECTION VIII

FAULT TRACKING

The table below presents the faults related to the RS-485 option board. For more information, see the ACCel500 Frequency Converters manual.

The RS-485 option board status LEDs have been described in more detail in section 3-6.

TABLE 8-1. RS-485 OPTION BOARD FAULTS

Fault Code	Fault	Possible Cause	Correcting Measures
37	Device change	Option board changed.	Reset
38	Device added	Option board added.	Reset
39	Device removed	Option board removed.	Reset
40	Device unknown	Unknown option board.	
53	Fieldbus fault	The data connection between the Modbus/ N2 Master and the RS-485 option board is broken	Check the installation. If installation is correct contact the nearest Avtron distributor.
54	Slot fault	Defective option board or slot	Check the board and slot. Contact the nearest Avtron distributor.

TABLE 8-2. FREQUENCY CONVERTER RESPONSES TO FAULTS

Code	Parameter	Min	Max	Unit	Step	Default	ID	Note
P2.7.22	Response to fieldbus fault	0	3		1	0	733	0 = No response 1 = Warning 2 = Fault,stop acc. to 2.4.7 3 = Fault,stop by coasting
P2.7.23	Response to slot fault	0	3		1	0	734	0 = No response 1 = Warning 2 = Fault,stop acc. to 2.4.7 3 = Fault,stop by coasting

---

APPENDIX A

A-1 PROCESS DATA OUT (SLAVE TO MASTER)

The Fieldbus Master can read the frequency converter's actual values using process data variables.

Basic, Standard, Local/Remote Control, Multi-Step Speed Control, PID control and Pump and fan control applications use process data as follows:

TABLE A-1. PROCESS DATA OUT VARIABLES

ID	Data	Value	Unit	Scale
2104	Process data OUT 1	Output Frequency	Hz	0.01 Hz
2105	Process data OUT 2	Motor Speed	rpm	1 rpm
2106	Process data OUT 3	Motor Current	A	0,1 A
2107	Process data OUT 4	Motor Torque	%	0,1 %
2108	Process data OUT 5	Motor Power	%	0,1 %
2109	Process data OUT 6	Motor Voltage	V	0,1 V
2110	Process data OUT 7	DC link voltage	V	1 V
2111	Process data OUT 8	Active Fault Code	–	–

The Multipurpose Control application has a selector parameter for every Process Data. The monitoring values and drive parameters can be selected using the ID number (see the ACCel500 Adjustable Frequency Drives Application manual for monitoring values and parameters). Default selections are as in the table above.

A-2 PROCESS DATA IN (MASTER TO SLAVE)

ControlWord, Reference and Process Data are used with all applications as follows:

TABLE A-2. PROCESS DATA OUT VARIABLES

ID	Data	Value	Unit	Scale
2003	Reference	Speed Reference	%	0.01%
2001	ControlWord	Start/Stop Command Fault reset Command	–	–
2004–2011	PD1 – PD8	Not used	–	–

TABLE A-3. MULTIPURPOSE CONTROL APPLICATION

ID	Data	Value	Unit	Scale
2003	Reference	Speed Reference	%	0.01%
2001	ControlWord	Start/Stop Command Fault reset Command	–	–
2004	Process Data IN1	Torque Reference	%	0.1%
2005	Process Data IN2	Free Analogia INPUT	%	0.01%
2006–2011	PD3 – PD8	Not Used	–	–

TABLE A-4. PID CONTROL AND PUMP AND FAN CONTROL APPLICATIONS

ID	Data	Value	Unit	Scale
2003	Reference	Speed Reference	%	0.01%
2001	ControlWord	Start/Stop Command Fault reset Command	–	–
2004	Process Data IN1	Reference for PID controller	%	0.01%
2005	Process Data IN2	Actual Value 1 to PID controller	%	0.01%
2006	Process Data IN3	Actual Value 2 to PID controller	%	0.01%
2007–2011	PD4–PD8	Not Used	–	–