D S 4
L A S E R T E C H N O L O G Y

COMMUNICATION PROTOCOL

DS4 GPB BOARD

Ver 2.02.07

Document version	1.01
Date	12/01/2009

DS4 S.r.l. Via Giardini, 32 - 24066 Pedrengo (BG) - Italy
Tel: 0039-35-661140 - Fax: 0039-35-658773 – www.ds4.it – info@ds4.it
Reg. Imprese Bergamo P.IVA. C.F. V.A.T.: IT 02477710160

SUMMARY:

1 SERIAL CONTROLS

In the serial communication the board assumes the **slave** rule, therefore it can reply only to the interrogation of a remote **master**.

All the protocol strings respect the following structure:

Master interrogation→ Slave:

INITIATOR	CONTROL	PARAMETERS	CRC	TERMINATOR

Positive Slave Reply→ Master :

INITIATOR	ACK	CONTROL	PARAMETERS	CRC	TERMINATOR

Negative Slave Reply→ Master :

INITIATOR	NACK	ERROR CODE	PARAMETERS	CRC	TERMINATOR

Initiator	= “COBS” constant
Control	= Consist in a byte, it states the meaning of the following parameters.
Parameters	= The contents of this field depend on the meaning of each single control and it has a variable length (from 0 to 34 bytes).
CRC	= Variable gained by the less significant byte of the XOR operation among all the string bytes but for the initiator.
Terminator	= Constant 0x00
Error code	= Consist in a byte, it describes the error in the given control. The description of the error codes is stated in chapter 3.

N.B.

In the reply string the fields ACK, NACK reply to ASCII standard, so ACK has value 6, while NACK has value 15.

The variables in the strings follow the Little Endian architecture, so the less significant byte is the first to be received. Example:

Variable 1 (16bit)		**Variable 2 (32 bit)**				**Variable 3 (8bit)**	**Variable 4 (32bit)**			
Byte 1 (LSB)	Byte 2 (MSB)	Byte 1 (LSB)	Byte 2	Byte 3	Byte 4 (MSB)	Byte 1	Byte 1 (LSB)	Byte 2	Byte 3	Byte 4 (LSB)

Before being transmitted, every string is codified through the COBS (**C**onsistent **O**verhead **B**yte **S**tuffing) algorithms, which is due to eliminate characters with the value equal to the terminator that are inside the sequence itself.

In the same way, after having received the string, it is necessary to apply the COBS decoding to read the original content.

The initiator and the terminator are excluded from the coding.

About the documentation concerning the COBS algorithm please refer to Consistent Overhead Byte Stuffing by Stuart Cheshire and Mary Baker.

N.B.

1) All the numeric values inside this document preceded by **0x** refers to hexadecimal values.

2) Appendix **D** – Communication parameters.

1.1 EEPROM READING

Returns the content of the EEPROM memory (Capacity 1 Kbyte) available on the board.

Interrogation: Master → Slave

INITIATOR	0x02	ADDRESS	N°Bytes	CRC	TERMINATOR

Reply: Slave → Master

INITIATOR	ACK	0x02	ADDRESS	CONTENT	CRC	TERMINATOR

During the interrogation the master must specify 2 fields:

ADDRESS: Consist in two bytes, state the position from which begin reading. It has values in the range 0 – 1023. If it is required a value outside this range, the slave will send a negative reply for wrong address.

N°Bytes: Specify how many bytes have to be read s tarting from the given address. It has values in the range 1 – 32. If it is required a value outside this range, the slave will send a negative reply for wrong parameters.

In the reply, the slave communicates once again the address, followed by the number of the required bytes.

N.B.

Reading a number of **16** bytes starting from the address **0** you get the serial number of the board.

The serial number is communicated in an ASCII extension, the first character of the string is the first byte of the **content** field, while the last one is those that preceded the 0x00 character inside of the communicated string. In this way the string has a maximum length of 15 characters.

1.2 VARIABLE WRITING

Every variable code is 2bytes.

Interrogation: Master → Slave

INITIATOR	0x0A	Variable	Value	CRC	TERMINATOR

Positive reply: Slave → Master

INITIATOR	ACK	0x0A	CRC	TERMINATOR

N.B.

Chapter 2 – Variables description.

1.3 VARIABLE READING

The dimension of the reply to this command depends on the required variable.

Every variable code is 2bytes.

.

Interrogation: Master → Slave

INITIATOR	**0x0B**	**Variable**	**CRC**	**TERMINATOR**

Reply: Slave → Master

INITIATOR	**ACK**	**0x0B**	**Variable**	**Value**	**CRC**	**TERMINATOR**

N.B.

Chapter 2 – Variables description.

2 VARIABLES DESCRIPTION:

2.1 Firmware Version:

Code: 0x0000
Name: GPB_VAR_FW_VER
Dimension: 3 bytes
Machinery : All
Description: Firmware version stated in the extension Version.Subversion.Revision
Each field can assume values from 0 to 255.

2.2 Protocol Version:

Code: 0x0002
Name: GPB_VAR_PROT_VER
Dimension: 3 bytes
Machinery : All
Description: Protocol version stated in the extension Version.Subversion.Revision
Each field can assume values from 0 to 255.

2.3 Present anomalies:

Code: 0x0100

Name: `GPB_VAR_ANOMALY`

Dimension: 4 bytes

Machinery : All

Description: Bit-to-bit mask concerning active anomalies.

The anomaly mask is 4 bytes, even if each machinery may have a different number of anomalies. The less significant bit corresponds to the first anomaly and so on, for a maximum of 32 anomalies.

The **1** level bits correspond to active anomalies.

N.B.

Appendix **C**: Anomaly codification.

2.4 IO status:

Code: 0x0200

Name: `GPB_VAR_IO_STATUS`

Dimension: 17 bytes

Machinery : All

Description: Status (On/Off) of all the I/O of the board.

Field	IN CPU	IN EXP	OUT CPU	OUT EXP	Relay
Byte	4	4	4	4	1

Each field represents bit per bit the status mask of the I/O of the associated section, the less significant bit corresponds to the first I/O and so on, for a maximum of 32 I/O for each machinery.

The value 1 bits correspond to the activated I/O.

Field description:

FIELD	Description
IN CPU	CPU board input – the first 12 bit of the field state the status, the other ones are unused.
IN EXP	Expansion board input – the first 24 bit of the field state the actual status. The more significant byte is unused.
OUT CPU	CPU board output – the first 12 bit of the field state the status, the other ones are unused.
OUT EXP	Expansion board output – The first byte contains the status of the 8 outputs Emitter/Bus while the second byte states the status of the 8 power outputs. The remaining 2 bytes are unused.
Relay	The bit 0-3 (less significant) states the status of the 4 relay outputs. The bit 4-5 state the status of the two led on the cpu board.

2.5 Analogical input:

Code: 0x0201

Name: `GPB_VAR_ANALOG_IN`

Dimension: 12 bytes

Machinery : All

Description: Value 10bit of the 6 analogical inputs.

Field	AN 0	AN 1	AN 2	AN 3	AN 4	AN 5
Byte	2	2	2	2	2	2

Each ANx field consists in 2 bytes, stating the analogical value of the associated input.

The fields are unsigned and can assume values in the range 0 – 1023.

According to the fact that the inputs have a resolution equal to 10bit and the maximum value of the referential tension is 5V, to covert the value from digital to analogical (stated in volt) it is necessary to apply the following formula:

$$\text{Value}_{(\text{Volt})} = \text{ANx} \times 5 / 2^{10}$$

2.6 16bit analogical output:

Code: 0x0202

Name: GPB_VAR_DAC16

Dimension: 2 bytes

Machinery : All

Description: DAC value 16 bit.

According to the fact that the output referential tension is in the range 0-10V, given the resolution of 16bit, to convert the value from digital to analogical it is necessary to apply the following formula:

$$Value_{(Volt)} = Value_{(dgt)} \times 10 / 2^{16}$$

N.B.

Doing the writing of this variable, you automatically activate the forcing. The active/deactivate status of the forcing is stated in the variable 0x020B.

2.7 Timekeeper time:

Code: 0x0300

Name: GPB_VAR_TK_TIME

Dimension: 3 bytes

Machinery: All

Description: Time available in the time keeper in format hh:mm:ss (0-24).

2.8 Timekeeper date:

Code: 0x0301

Name: GPB_VAR_TK_DATE

Dimension: 4 bytes

Machinery : All

Description: Date set in the time keeper in format dd:mm:yy as well as the day of the week.

Field	Day	Month	Day of the week (1-7)	Year
Byte	1	1	1	1

2.9 MULTIH head number:

Code: 0x0800

Name: GPB_VAR_MULTIH_HEADS_NUM

Dimension: 1 byte

Machinery: Multi-head machinery

Description: States the number of remote heads connected to the master (minimum value 1 , maximum value 40).

2.10 MULTIH Heads in communication timeout:

Code: 0x0810

Name: GPB_VAR_MULTIH_COM_TOUT

Dimension: 4 byte

Machinery: Multi-head machinery

Description: Bit mask stating remote heads in communication timeout anomaly.

The less significant byte states the head n. 1

After reception this anomaly is recommended to check the bus connections.

2.11 MULTIH Heads in position error:

Code: 0x0811

Name: GPB_VAR_MULTIH_POS_ERR

Dimension: 4 byte

Machinery: Multi-head machinery

Description: Bit mask stating remote heads in anomaly because of a wrong positioning of the cylinder.

The less significant byte states the head n. 1

After reception this anomaly is recommended to check the state of cylinder and the state of compressed air on the head having the problem.

2.12 MULTIH Heads in CRC error:

Code:	0x0812
Name:	`GPB_VAR_MULTIH_CRC_ERR`
Dimension:	4 byte
Machinery:	Multi-head machinery
Description:	Bit mask stating remote heads in anomaly because of CRC error. The less significant byte states the head n. 1 After reception this anomaly is recommended to check the bus connection or the state of the slave board.

3 ERROR CODE:

The error codes follow the NACK character in the negative reply string and their purpose it's to clarify the error situation to the master.

Negative Slave Reply String→ Master :

INITIATOR	NACK	ERROR CODE	PARAMETERS	CRC	TERMINATOR

CODE	PARAMETERS	DESCRIPTION
1	None	Initiator not found inside the received string.
2	None	Wrong CRC.
3	None	Unknown or unsupported control.
4	None	Parameters not valid.
5	Number of the unknown variable (2 bytes)	Nonexistent variable.
6	Number of the required variable (2 bytes)	Attempt to write a read only variable.
7	Wrong address (2 bytes)	EEPROM required address not valid.
8	None	EEPROM writing attempt across two pages.
9	Value communicated into the writing control (2 bytes), followed by the value read as verification of the writing (2 bytes).	Writing failed on DAC 16bit.
10	None	EEPROM previous writing still in process.
11	None	Not admissible control during the current working phase.
12	None	Security code for protected controls not valid or not communicated.

APPENDIX A: Protocol controls

Control	Code	Parameters [bytes]	Machinery kind	Introduction version
EEPROM reading	0x02	Address [2] + bytes number [1]	all	1.00.00
Variable writing	0x0A	Variable number [2] + Variable form [1-64]	all	2.00.00
Variable reading	0x0B	Variable number [2]	all	2.00.00

APPENDIX B: protocol variables

Number	Name	Description	Dimension (Byte)	Reading only	Protected writing	Machinery kind	Introduction varsion
0x0000	GPB_VAR_FW_VER	Firmware version (Ver.SubVer.Rev)	3	x		All	2.00.00
0x0001	GPB_VAR_MACHINE	Machinery kind	2	x		All	2.00.00
0x0002	GPB_VAR_PROT_VER	Protocol version GPB supported (Ver.SubVer.Rev)	3	x		All	2.00.00
0x0100	GPB_VAR_ANOMALY	Anomaly mask	4	x		All	2.00.00
0x0200	GPB_VAR_IO_STATUS	On/off status of all the I/O of the board	17	x		All	2.00.00
0x0201	GPB_VAR_ANALOG_IN	Analogical input value	12	x		All	2.00.00
0x0202	GPB_VAR_DAC16	Analogical output value 16bit	2		x	All	2.00.00
0x0300	GPB_VAR_TK_TIME	Time keeper time in format hh:mm:ss (0-24)	3			All	2.00.00
0x0301	GPB_VAR_TK_DATE	Time keeper date in format dd:mm:Week Day:yy	4			All	2.00.00
0x0800	GPB_VAR_MULTIH_HEADS_NUM	Configured remote head number	1			MULTIH	2.02.05
0x0810	GPB_VAR_MULTIH_COM_TOUT	Remote heads in timeout anomaly	4			MULTIH	2.02.05
0x0811	GPB_VAR_MULTIH_POS_ERR	Remote heads in wrong positioning anomaly	4			MULTIH	2.02.05
0x182	GPB_VAR_MULTIH_CRC_ERR	Remote heads with CRC error	4			MULTIH	2.02.07

APPENDIX C: Anomaly codification

Here below there are the anomaly codifications for each DS4 machinery interfaced with a GPB board.

C7 – Multi-head machinery:

Bit	Description
1	Emergency pressed
2	Reset
3	Laser not ready
4	Laser temperature
5	Laser voltage
6	Start error (reset start before reset pulse)
7	Power fail on a CPU board
8	ID Error. (Selected an ID out of range or an ID previously excluded)
9	Head Not Ready (Check the head cylinder)
10	Reply timeout from one or more remote heads(Check variable `GPB_VAR_MULTIH_COM_TOUT`)
11	Cylinder positioning error on one or more heads (Check variable `GPB_VAR_MULTIH_POS_ERR`)
12	CRC error (Check variable `GPB_VAR_MULTIH_CRC_ERR`)
13	Shutter position error
14 → 32	Unused

APPENDIX D: Serial parameter communication

Here below there are the communication parameters used on the machineries managed by a GPB board:

Machinery	Parameters
Multi-head	9600,N,8,1