



EVO UPLOAD/DOWNLOAD PROTOCOL

PROJECT : *UPLOAD/DOWNLOAD PROTOCOL*
MODULE : *COMMUNICATION*
ENGINEERS : *JEAN-FRANCOIS POIRIER / STÉPHANE BOLDUC/ERIC LARIVIÈRE*
FILE NAME : *EVO_UDSProtocol02.doc*
LAST UPDATE: *Wednesday, September 26, 2007*

FUNCTIONAL DESCRIPTION

This protocol is used to communicate between a PC through a modem /serial port and a control panel, a Digiprint or a Contact module.

On site programming can be done with the PC using the PC serial port and the module's serial port.

All modules on bus can be up/downloaded through the panel or the Digiprint module.

A CRC checksum is used to verify the data integrity when communicating through phone line & a simple Checksum is used when doing direct connection (addition of all the bytes).

CONTENTS

- [FEATURES](#)

- [UNRESOLVED ISSUES](#)

- [PROTOCOL OPERATION](#)

- [LIST OF CHANGES](#)

- **LIST OF COMMANDS**

[PC](#)

0x00	<u>Initialization</u>
0x10	N/U
0x20	<u>Optimize Modem Speed</u>
0x30	<u>Set panel time</u>
0x40	<u>Monitoring command</u>
0x50	<u>Read (upload)</u>
0x5F	<u>Serial Connection Initialization</u>
0x60	<u>Write (download)</u>
0x70	<u>Disconnect command</u>
0x72	<u>Initialize UIP Programming</u>
0x73	<u>Request Firmware Upgrade</u>
0x74	<u>Request Panel ID</u>
0x7F	<u>Selftest</u>
0x80	<u>Save last uploaded event number</u>
0x90	<u>Save / read user information</u>
0xA0	<u>General send command</u>
0xB0	<u>Message Save & broadcast</u>
0xC0	<u>General Command (Extended 512 Bytes)</u>
0xC0	Unlock installation (N/U)
0xD0	<u>Zone status change</u>
0xE0	<u>Request Event number xxxx from buffer</u>

Controller

0x00	<u>Initialization</u>
0x10	<u>Login confirmation + call back</u>
0x20	<u>Optimize modem speed</u>
0x30	<u>Set panel time confirmation</u>
0x40	<u>Monitoring command confirmation</u>
0x50	<u>Read (upload)</u>
0x60	<u>Write (download)</u>
0x70	<u>Panel Error Message</u>
0x74	<u>Panel ID Request Answer</u>
0x80	<u>Save last uploaded event number confirmation</u>
0x90	<u>Save / read user information</u>
0xA0	<u>General send command confirmation</u>
0xB0	<u>Message broadcast confirmation</u>
0xB0	<u>Bus Data</u>
0xC0	<u>General Command (Extended 512 Bytes)</u>
0xC0	Unlock installation confirmation (N/U)
0xD0	<u>Zone status change confirmation</u>
0xE0	<u>Event transmission and requested event</u>

FEATURES

- 300 BPS speed communication using reverse frequencies (via modem).
- 38400 Baud / 57600 Baud using direct connection (serial port) (8 bits / 1 stop bit / no parity).
- The PC always uses a mark frequency (1) of 2225Hz and a space frequency (0) of 2025Hz.
- The module always uses a mark frequency (1) of 1270Hz and a space frequency (0) of 1070Hz.
- Panel status monitoring in real time.
- Any module connected on the bus can be up/downloaded (programmed).
- **The communication is 37 bytes for commands:**
 - **0x5F → Serial Initialization**
 - **0x00 → Login Command**
 - **0x72 → Univesral Infield Programming Command**
- **The communication is variable size (512 bytes + header) for commands:**
 - **0xC- → General Command**
- **The communication si variable size (64 bytes + header) for all other commands.**
- Events are sent on the serial port as they occur (no connection needed).

UNRESOLVED ISSUES

- When an alarm occurs, the PC software has to hang up the communication if the reporting is enabled. The panel sends a bit to tell the PC software that the panel has to report an alarm (Implemented) (No need on Digiplex NE).

PROTOCOL OPERATION

Note: In that document the word “panel” signifies the control panel or the Digiprint and “PC” signifies the computer. The words upload and download are used and means respectively read and write.

- The module can answer a phone call or can directly call the PC. Upon a phone call, the module first generates its own carrier of 1270Hz for 16 seconds if calls or 8 seconds if answers, then it sends the initialization string which contains the product ID, the software version and the module identifier code with a Checksum on top of that.
- The PC’s modem should recognize the carrier & should connect to the module. A 2225Hz carrier is emitted on top of the module’s carrier.
- After the module sent the initialization string, the PC verifies the integrity of the whole string using the Checksum and verifies the “module identifier code”, software version and the product ID.
- If the module identifier corresponds to the one programmed in the PC, the PC sends back the data that initially sent by the module including its PC password.
- Then the module verifies the data integrity and the PC password, if it’s the same as the one programmed in the module, the login is completed.
- From there, the PC takes control of the whole communication.
- The panel/module status is sent to the PC, and then the module waits for an upload or download command from the PC.

• **WHAT CONTAINS THE MODULE INITIALIZATION STRING?**

1- 37 bytes are sent where:

Byte 00 → First nibble is the command and the second nibble is the panel special message

Byte 01 → Module's address (module's address correspond to the address on the bus)

Byte 02 and 03 → Eeprom address (15 bits) from 0000 to 7FFF. Bit 15 (Bit 7 of byte 3) is the destination/source type (0 = EEPROM, 1 = RAM)

Byte 04 → Product ID (00 = Digiplex / 10 = Spectra / 20 = Contact)

- 0x03 = EVO 48
- 0x04 = EVO 96
- 0x05 = EVO 192

Byte 05 → Marketing Software Version

Byte 06 → Marketing Software Revision

Byte 07 → Marketing Customer ID

Byte 08 and 09 → Module Identifier

Byte 10 and 11 → PC Password (Reserved)

Byte 12 → Winload ID according Product ID

Byte 13 → Memory Map Version / Revision

Byte 14 → Event List Version / Revision

Byte 15 and 16 → Firmware Build Number

Byte 17 to 20 → Module's serial number

Byte 21 to 35 → Sections [3030] to [3038] (EVO)

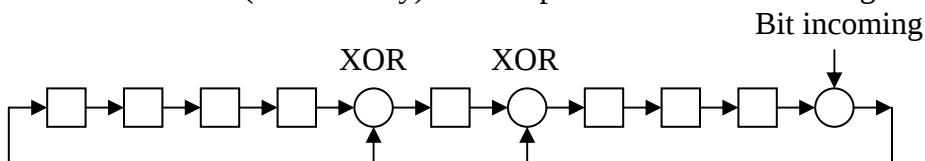
Byte 36 → Checksum

Example:

NB:	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18...	32	33	34	35	36
TX:	00	00	01	EF	00	00	01	00	12	34	00	00	05	13	25	B0	12	00	00...	00	00	00	00	A6
DT:	--	--	--	--	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15...	29	30	31	32	--

Details: Modules sends Initialization string with Module ID read at address 0x1EF / Product ID 0x00 / software versions 00.01.00 / panel ID 1234 / Winload ID 0x05 / Memory MAP Version 1 Revision 3 / Event List Version 2 Revision 5 / Firmware version B012 & checksum 0xA6.

CRC checksum (modem only) is a complex calculation following this:



- Each bit of a byte coming is XOR with the bit 0 of CRC check register, the result of the XOR is used to change the bit 4 and 3 with a XOR as well then all bits are shifted to the right for the next bit.

- **WHAT PC ANSWERS TO THAT?**

The PC checks the checksum and the Module identifier, if everything is ok, a feedback string is sent to the module. Where the data is the same as the module sent plus the PC password.

1- 37 bytes are sent where:

Byte 00 → First nibble is the command and the second nibble is the panel special message

Byte 01 → Module's address (module's address correspond to the address on the bus)

Byte 02 and 03 → Empty (00)

Byte 04 → Product ID (00 = Digiplex / 10 = Spectra / 20 = Contact)

- 0x03 = EVO 48
- 0x04 = EVO 96
- 0x05 = EVO 192

Byte 05 → Marketing Software Version

Byte 06 → Marketing Software Revision

Byte 07 → Marketing Customer ID

Byte 08 and 09 → Module identifier

Byte 10 and 11 → PC password

Byte 12 → Communication speed (number of carrier bits before the data packet [0x00 to 0x0F])

Byte 13 → Communication Type

Byte 14 to 16 → User Code

Byte 17 to 34 → Empty (00)

Byte 35 → Communication speed (number of carrier bits after the data packet [0x00 to 0xFF])

Byte 36 → Checksum

Example:

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22
TX: 00 00 01 EF 00 00 01 00 12 34 56 78 10 00 00 00 00 00 00 00 00 00 00
DT: -- -- -- -- 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19
```

```
NB: 23 24 25 26 27 28 29 30 31 32 33 34 35 36
TX: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 6E
DT: 20 21 22 23 24 25 26 27 28 29 30 31 32 --
```

Details: PC sends back initialization string + PC password 5678 + communication speed 0x10 and checksum 6E.

Then the module verifies the data integrity with checksum, verifies also the PC password sent. It as to be the same as the one programmed in the module memory.

If there's an error, either the checksum or the PC password is not correct, the module sends again the initialization string. This sequence is repeated up to 3 times and then the module hangs up.

If PC password and checksum are correct, the module sends a confirmation to the PC.

1 - 6 bytes (or 37) are sent where:

Byte 00 → First nibble is the command and the second nibble is the panel special message

Byte 01 → Number of bytes in packet.

Byte 02 → Bit 4 (Winload Answers (0) / Neware Answers (1))

Byte 03 and 04 → User Number / Callback Enable

Byte 04 → User Assignment

Byte 05 → Checksum

Example 1:

NB: 00 01 02 03 04 05

TX: 10 00 00 00 00 38 → call back is not enabled

Example 2:

NB: 00 01 02 03 04 05

TX: 10 00 FF FF 00 8C → call back is enabled

If call back is enabled, the module sends that string then hang up to call back the PC. After the call back, the same sequence, from the beginning, is executed except that the call back is not re-executed.

- **Then the PC upload (reads) the module status.**

1 - 8 bytes (or 37) are sent where:

Byte 00 → First nibble is the command and the second nibble is the panel special message

Byte 01 → Number of bytes in the packet

Byte 02 → Control byte

Byte 03 → Module's address (module's address correspond to the address on the bus)

Byte 04 and 05 → Eeprom starting address (16 bits) from 0000 to FFFF / RAM block number (panel) . (Bit 7 of byte 2) is the destination/source type (0 = EEPROM, 1 = RAM) (Note: in this case this bit is set.)

Byte 06 → Number of bytes to read

Byte 07 → Checksum

Example 1:

NB: 00 01 02 03 04 05 06 07

TX: 50 08 80 00 00 01 40 19

Details: PC sends command 5 to read 64 bytes of the module 00 (in this case 00 is the control panel) RAM block 01 (hex) and the checksum is 0x19 .

The module verifies the integrity of that string with the Checksum and sends the information to the PC.

2 - 71 bytes are sent where:

Byte 00 → First nibble is the command and the second nibble is the panel special message
Byte 01 → Number of bytes in packet
Byte 02 → Module's address (module's address correspond to the address on the bus)
Byte 03 → Control Byte (Bit 7 = EEPROM/RAM)
Byte 04 and 05 → EEPROM Address / RAM block
Byte 06 to 69 → Data bytes
Byte 70 → Checksum

Example 1:

```
NB: 00 01 02 03 04 05 06 07 ... 69 70  
TX: 50 47 00 80 00 01 DD DD ... DD CC
```

Details: In this example the control panel return the same command, module's address and RAM address. The data bytes contains the first block of ram from the panel.

The PC reads, as many bytes as it needs to get the whole needed module information.

- **From there, the PC user can do any action to send or read any information of any modules.**
- **An upload of the Eeprom is done the same way.**
- **In case of download (write), the PC sends up to 71 bytes containing the data to write in the Eeprom.**

1 - 71 bytes are sent where:

Byte 00 → First nibble is the command and the second nibble is the special message
Byte 01 → Number of bytes in packet
Byte 02 → Control byte
Byte 03 → Module's address (module's address correspond to the address on the bus)
Byte 04 and 05 → Starting address
Byte 06 to 69 → Data bytes
Byte 70 → Checksum

Example 1:

```
NB: 00 01 02 03 04 05 06 07 ... 69 70  
TX: 60 47 00 00 00 10 DD DD ... DD CC
```

Details: In this example the control PC sends the download command, module's address and Eeprom address. 64 bytes of data are written starting from address 0x10.

- **Then the module answers with 7 bytes to tell the PC that the message is understood.**

1 - 7 bytes are sent where:

Byte 00 → First nibble is the command and the second nibble is the panel special message

Byte 01 → Number of bytes in packet

Byte 02 → Module's address (module's address correspond to the address on the bus)

Byte 03 → Control byte

Byte 04 and 05 → Starting address

Byte 06 → Checksum

Example 1:

NB: 00 01 02 03 04 05 06

TX: 60 07 00 00 00 10 77

CHANGES FROM PREVIOUS VERSION

- Added a bit that tells if the panel is in alarm (and not reported) (26/08/99).
- Updated all the commands (for the message bit) (01/09/99).
- Detailed all module & PC commands (09/09/99).
- Added Information in initialization string such as WinLoad ID, Firmware Version, Memory Map ID and Event List ID (28/06/00).
- Added event transmission on serial port only. Command D (30/10/00).
- Added 2 flags in the message center to indicate if the end user software is connected or not.
- Added a new command for event buffer events transfer.
- Added Information in initialization string such as Module's serial number (20/04/01).
- Added Information in initialization string such as Digiplex NE section 3030 to 3038 (08/05/01).
- Added Information in Action failed command as invalid code or partition in lockout (08/05/01).
- Added the list of events for the serial port (09/08/01).
- Added support for 1Mb for GLCD (28/01/02).
- Added a command so Winload / Neware can locate a graphic LCD keypad (02/05/02).
- Added a value used as an address offset in read data from module command 0x5? (02/05/02).
- Added explanation for serial port connection (need to send a connection string before the panel sends its initialization string) (13/08/02).
- Added support for 192 zones for future version of panel (11/03/03).
- Added a modem parameter to support more modems (number of carrier bits after a data packet sent by the panel in commands 0 (Initialisation) and 2 (set modem speed) (06/17/03).
- Added support for 15 partitions in most of the commands for future expansion (06/17/03).
- Updated document for the serial port (always 37 bytes) (07/05/04).
- Added a command for serial disconnection (0x70 – message 5) (08/01/05)
- Added command 0x73 to support the panel firmware upgrade through Winload or NEWare. This is used with serial connection only. (08/01/05)
- Modified the protocol to have it variable in length in almost every commands. Also added the possibility to address 254 modules and up to 256Kbytes of eeprom address.
- Modified the event transmission commands. (November 29, 2005)
- Added a command to do a self test with testing jig. (January 23, 2006)
- Added new information for command 0x72. (July 5, 2007)
- Added command 0xC0 with possible 512 bytes of data. (July 5, 2007)
- Added bit 4 in serial event to tell ip100 not to report an event. (August 16, 2007)
- Added label in serial event with label type. (August 16, 2007)
- Added Source ID in PC initialization command to tell panel if software is connected through IP/GSM/Direct. (August 16, 2007)
- Added error code 0x1C to tell Winload/NeWare that it could not write a label because the label number is out of range (Invalid Label Number). (August 16, 2007)

LIST OF COMMANDS (PC to Panel)

Message: Initialization.

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22...34 35 36
TX: P- AA -- -- II SS RR CC MM MM WW WW DD TT UU UU UU NN NN NN NN DD DD...SC SO VV
DT: -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18...-- -- --
```

Where:	P	PC Command: 0	1 nibble
	A	Module address (00= panel/module)	1 byte
	I	Product ID	1 byte
		0x00 = Digiplex (Version 1.30 & below)	
		0x01 = Digiplex (Version 2.00 & up)	
		0x02 = Digiplex NE (96 zones 8 partitions 999 users)	
		0x03 = Digiplex EVO (48 zones 4 partitions 256 users)	
		0x04 = Digiplex EVO (96 zones 8 partitions 999 users)	
		0x05 = Digiplex EVO (192 zones 8 partitions 999 users)	
	S	Software version	1 byte
	R	Software revision	1 byte
	C	Software ID	1 byte
	M	Module ID	1 byte
	W	PC Password	1 byte
	D	Modem speed	1 byte
	T	Winload Type ID	1 byte
		0x55 = end user software	
		0x56 = end user software using end user PC password (TCP/IP)	
		Any other value = Winload	
	U	User code (BCD format)	6 nibbles
		Only the end user software sends this.	
	N	Module's Serial Number	8 nibbles
	D	EVO: Section 3030 to 3038 data	9 bytes
	SC	Source ID	1 byte
		0x00 Non-Valid Source	
		0x01 Winload Direct	
		0x02 Winload Through IP	
		0x03 Winload Through GSM	
		0x04 Winload Through Dialer	
		0x05 NeWare Direct	
		0x06 NeWare Through IP	
		0x07 NeWare Through GSM	
		0x08 NeWare Through Dialer	
		0x09 IP Module Direct	
		0x0A VDMP3 Direct	
		0x0B VDMP3 GSM	
	S0	Carrier length	1 byte
		(Number of carrier bits after data packet)	
	VV	Checksum	1 byte

Message: Optimize modem speed.

NB: 00 01 02 03 04 05
TX: P- BB S0 S1 S2 VV
DT: -- -- -- -- -- --

Where:	P	PC Command: 2	1 nibble
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	S0	Carrier length (Number of carrier bits after data packet)	1 byte
	S1	Modem speed 1	1 byte
	S2	Modem speed 2 (has to be the same as 1)	1 byte
	VV	Checksum	1 byte

Message: Set panel time.

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12
TX: P- BB TT -- -- -- CC YY MM DD HH MM VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 --

Where:	P	PC Command: 3	1 nibble
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	TT	Bus address	1 byte
		0x00 = Panel	
		0x01 to 0xFE = Modules	
	C	Century	1 byte
	Y	Year	1 byte
	M	Month	1 byte
	D	Day	1 byte
	H	Hour	1 byte
	M	Minute	1 byte
	VV	Checksum	1 byte

Message: Monitoring command

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14
TX: P- BB -- -- -- -- 12 34 56 78 -- -- -- -- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 --

Where:	P	PC Command: 4	1 nibble
	1	Partition 01 command (see list below)	1 nibble
	2	Partition 02 command (see list below)	1 nibble
	3	Partition 03 command (see list below)	1 nibble
	4	Partition 04 command (see list below)	1 nibble
	5	Partition 05 command (see list below)	1 nibble
	6	Partition 06 command (see list below)	1 nibble
	7	Partition 07 command (see list below)	1 nibble
	8	Partition 08 command (see list below)	1 nibble
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	VV	Checksum	1 byte

List of commands:

0:	No effect
1:	Future use
2:	Full arm partition
3:	Stay arm partition
4:	Instant arm partition
5:	Force arm partition
6:	Disarm partition
7:	Future use
8:	Beep all keypads of partition
9:	Future use
A:	Future use
B:	Future use
C:	Future use
D:	Future use
E:	Future use
F:	Future use

Message: Upload (read) module's Eeprom/RAM data from address HHLL.

NB: 00 01 02 03 04 05 06 07
TX: PN BB CC TT HH LL UU VV
DT: -- -- -- -- -- -- --

Where:	P	PC command: 5	1 nibble
	N	Block number	
		0: from 0x00000 to 0x0FFFF	
		1: from 0x10000 to 0x1FFFF	
		...	
		F: from 0xF0000 to 0xFFFFF	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	CC	Control Byte	8 bits
		Bit 7: 0 = Eeprom access / 1 = RAM access	
		Bit 6: Alarm reporting pending	
		Bit 5: Winload software is connected	
		Bit 4: NEware software is connected	
		Bit 3: Future use	
		Bit 2: Future use	
		Bit 1: Eeprom address bit 17	
		Bit 0: Eeprom address bit 16	
	TT	Bus address	1 byte
		0x00 = Panel	
		0x01 to 0xFE = Modules	
	HH	Eeprom/RAM Address high	
	LL	Eeprom/RAM Address low	1 byte
	UU	Number of bytes to read	1 byte
	VV	Checksum	1 byte

Message: SERIAL CONNECTION INITIALISATION:

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21...35 36
TX: P- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- VV
DT: 5F 20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00...00 7F
```

Note : In order to initialize a serial connection with a module, this string must be sent. This string will make the panel/module sends its initialization string as per command 0x00.

Message: Download (write) data into module's Eeprom/RAM at address HHLL.

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21...69 70
TX: PN BB CC TT HH LL DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD...DD VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15...63 --

Where: P PC command: 6 1 nibble
N Block number
0: from 0x00000 to 0x0FFFF
1: from 0x10000 to 0x1FFFF
...
F: from 0xF0000 to 0xFFFFF
BB Number of bytes in packet (MAXIMUM 71) 1 byte
CC Control Byte 8 bits
Bit 7: 0 = Eeprom access / 1 = RAM access
Bit 6: Alarm reporting pending
Bit 5: Winload software is connected
Bit 4: NEware software is connected
Bit 3: Future use
Bit 2: Future use
Bit 1: Eeprom address bit 17
Bit 0: Eeprom address bit 16
TT Bus address 1 byte
0x00 = Panel
0x01 to 0xFE = Modules
HH Eeprom/RAM Address high
LL Eeprom/RAM Address low 1 byte
DD Data bytes (up to 64 bytes) 64 bytes
VV Checksum 1 byte

Message: Terminate connection. (serial connection only)

NB: 00 01 02 03 04 05 06
TX: P- BB MM -- -- -- VV
DT: -- -- -- -- -- -- --

Where:	P	Panel Command: 0x70	1 byte
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	MM	Message to Winload:	1 byte
		0x00 to 0x04 : Future use	
		0x05: Disconnect	
		0x06 to 0x0F : Future use	
	VV	Checksum	1 byte

Note: The PC sends this command with the message 0x05 to terminate the connection with the panel. The panel sends the same command as answer and then terminate the connection.

Message: Initiate communication using UIP software. (serial connection only)

PC:

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...37
TX: PP -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- VV
DT: -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19...--
```

PANEL ANSWER:

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
TX: PP ZZ ID VV RR BB FF II TT V1 V2 V3 S1 S2 S3 S4 H1 H2 B1 B2 B3 B4 B5 B6 B7
DT: -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20
```

```
NB: 25 26 27 28 29 30 31 32 33 34 35 36
TX: PP CC EE -- -- -- -- -- -- -- -- VV
DT: 21 22 23 24 25 26 27 28 29 30 31 --
```

Where:

PP	Panel Command: 0x72	1 byte
ZZ	New protocol : 0xFF	1 byte
ID	Protocol ID : Winload : 0	1 byte
VV	Protocol version	1 byte
RR	Protocol revision	1 byte
BB	Protocol build	1 byte
FF	Familly ID	1 byte
II	Product ID	1 byte
TT	Talker : 0 -> Boot loader	1 byte
	1 -> Controller application	
	2 -> Module aplication	
V1	Application version	1 byte
V2	Application revision	1 byte
V3	Application build	1 byte
S1-S4	Serial Number	4 bytes
H1	Hardware version	1 byte
H2	Hardware revision	1 byte
B1	Bootloader version	1 byte
B2	Bootloader revision	1 byte
B3	Bootloader build	1 byte
B4	Bootloader Date	1 byte
B5	Bootloader Month	1 byte
B6	Bootloader Century	1 byte
B7	Bootloader Year	1 byte
PP	Processor ID	1 byte
CC	Checksum ID	1 byte
EE	Encryption ID	1 byte
VV	Checksum	1 byte

Message: Request for firmware upgrade. (serial connection only)

PC:

NB: 00 01 02 03 04 05 06 07 08 09 10
TX: PP BB -- -- -- -- D0 D1 -- D3 VV
DT: -- -- -- -- -- -- 00 01 02 03 --

PANEL ANSWER:

NB: 00 01 02 03 04 05 06 07 08
TX: PP BB -- -- -- -- D0 D1 VV
DT: -- -- -- -- -- -- 00 01 --

Where:	PP	Panel Command: 0x73	1 byte
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	D0	0x55 : key 0	1 byte
	D1	0xAA : key 1	1 byte
	D3	Encryption ID	
	VV	Checksum	1 byte

Note: After the panel sent the answer it restart the software to 0 and then enter in boot loader execution.

Message: Panel id request (N/U → 0x72 contains same info except panel's version date)

PC:

NB: 00 01 02 03 04 05 06
TX: PP BB -- -- -- -- VV
DT: -- -- -- -- -- -- --

PANEL ANSWER:

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22...70
TX: PP BB -- -- BH BL BD BM BC BY PH PL PB PD PM PC PY S1 S2 S3 S4 EE -- VV
DT: -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18...--

Where:	PP	Panel Command: 0x74	1 byte
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	BH	Boot-Loader version high	1 byte
	BL	Boot-Loader version low	1 byte
	BD	Boot-Loader date	1 byte
	BM	Boot-Loader month	1 byte
	BC	Boot-Loader century	1 byte
	BY	Boot-Loader year	1 byte
	PH	Panel version high	1 byte
	PL	Panel version low	1 byte
	PB	Panel version build	1 byte
	PD	Panel date	1 byte
	PM	Panel month	1 byte
	PC	Panel century	1 byte
	PY	Panel year	1 byte
	Sx	Panel serial number	4 bytes
	EE	Encryption ID	1 byte
	VV	Checksum	1 byte

Message: self test from testing jig

PC:

NB: 00 01 02 03 04 05 06 07 08 09...70

TX: PP BB -- -- D0 D1 D2 D3 D4 D5...VV

DT: -- -- -- -- -- -- -- -- -- -- --...

Where:	PP	Panel Command: 0x7F	1 byte
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	Dx	Command receive from testjig for selftest	1 byte
	VV	Checksum	1 byte

Message: Save last uploaded event number.

PC:

NB: 00 01 02 03 04 05 06

TX: P- BB -- -- NN NN VV

DT: -- -- -- -- -- -- --

Where:	P	PC Command: 8	1 nibble
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	N	Event number (16 bit pointer)	4 nibbles
	VV	Checksum	1 byte

Message: Save/read user information.

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22...69 70
TX: PK BB -- NN HH LL D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE D0 D1...-- VV
DT: -- -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...64 --

Where:	P	PC command: 9	1 nibble
	K	Sub-command	1 nibble
		0 = Save	
		1 = Read	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	NN	Number of user to save or read (up to 4)	1 byte
	HH	User number high (First user)	1 byte
	LL	User number low (First user)	1 byte
	DD	User Data	64 bytes
	VV	Checksum	1 byte

Note : When saving the user information is sent by the PC in data bytes.
When reading the command has 7 bytes long.

Message: General command (general description).

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...70
TX: PK BB TT -- HH LL DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD -- -- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63 --

Where: P PC command: A 1 nibble
 K Sub-command 4 bits
 Bit 3: 0= Bus Address 1= Module Type
 Bit 2-0: 0: General broadcast
 1: LCD message off
 2: LCD message on low priority
 3: LCD message on high priority
 4: PGM command
 5: Start scan/remove module
 6: Un/Locate Graphic LCD keypad
 7: Start Bitmap transfer
BB Number of bytes in packet **(MAXIMUM 71)** 1 byte
TT Bus address / Serial number 1 byte
HH Address high / partition aimed 1 byte / 8 bits
LL Address low / partition aimed 1 byte / 8 bits
DD Data bytes (up to 64 bytes) 64 bytes
VV Checksum 1 byte

Note: Data byte numbers 16 to 63 are empty because of the bus protocol limitation when sending bus command.

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...70
TX: PK BB TT  -- HH LL DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD -- -- VV
DT:  --  --  --  --  --  -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63  --
```

Note: Data byte numbers 16 to 63 are empty because of the bus protocol limitation.

Message: General command sub command: LCD message OFF.

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...70
TX: PK BB TT -- HH LL DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD -- -- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63 --

Where:

P	PC command: A	1 nibble
K	Sub-command	4 bits
	Bit 3: 0= Bus Address 1= Module Type	
	Bit 2-0: 0x1: LCD message OFF	
BB	Number of bytes in packet (MAXIMUM 71)	1 byte
TT	Bus address / Serial number	1 byte
HH	partition aimed	
	Bit 0: Partition 1	
	Bit 1: Partition 2	
	Bit 2: Partition 3	
	Bit 3: Partition 4	
	Bit 4: Partition 5	
	Bit 5: Partition 6	
	Bit 6: Partition 7	
	Bit 7: Partition 8	
LL	Future use	
DD	Data bytes (up to 64 bytes)	64 bytes
VV	Checksum	1 byte

Note: Data byte numbers 16 to 63 are empty because of the bus protocol limitation.

Message: General command sub command: LCD message LOW priority.

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...70
TX: PK BB TT -- HH LL DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD -- -- VV
DT: -- -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63 --

Where: P PC command: A 1 nibble
K Sub-command 4 bits
Bit 3: 0= Bus Address 1= Module Type
Bit 2-0: 0x2: LCD message LOW priority
BB Number of bytes in packet (MAXIMUM 71) 1 byte
TT Bus address / Serial number 1 byte
HH partition aimed
Bit 0: Partition 1
Bit 1: Partition 2
Bit 2: Partition 3
Bit 3: Partition 4
Bit 4: Partition 5
Bit 5: Partition 6
Bit 6: Partition 7
Bit 7: Partition 8
LL Future use
DD Data bytes (up to 64 bytes) 64 bytes
VV Checksum 1 byte

Note: Data byte numbers 16 to 63 are empty because of the bus protocol limitation.

Message: General command sub command: LCD message HIGH priority.

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...70
TX: PK BB TT -- HH LL DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD -- -- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63 --

Where: P PC command: A 1 nibble
K Sub-command 4 bits
Bit 3: 0= Bus Address 1= Module Type
Bit 2-0: 0x3: LCD message HIGH priority
TT Bus address / Serial number 1 byte
BB Number of bytes in packet (MAXIMUM 71) 1 byte
TT Bus address / Serial number 1 byte
HH partition aimed
Bit 0: Partition 1
Bit 1: Partition 2
Bit 2: Partition 3
Bit 3: Partition 4
Bit 4: Partition 5
Bit 5: Partition 6
Bit 6: Partition 7
Bit 7: Partition 8
LL Future use
DD Data bytes (up to 64 bytes) 64 bytes
VV Checksum 1 byte

Note: Data byte numbers 16 to 63 are empty because of the bus protocol limitation.

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...70
TX: PK BB TT -- -- -- D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE DF --...-- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63 --
```

Where:	P	PC command: A	1 nibble
	K	Sub-command	4 bits
		Bit 3: 0= Bus Address 1= Module Type	
		Bit 2-0: 0x4: PGM override	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	TT	Bus address / Serial number	1 byte
	D0	PGM01 status (see list below)	1 byte
	D1	PGM02 status (see list below)	1 byte
	..		
	DF	PGM16 status (see list below)	1 byte
	VV	Checksum	1 byte

```
PGM status:      00:  No change on PGM
                  01:  Override PGM OFF
                  02:  Override PGM ON
                  03:  Release PGM OFF
                  04:  Release PGM ON
```

Note: Data byte numbers 16 to 63 are empty because of the bus protocol limitation.

Message: Bus action

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...70
TX: PK BB TT -- -- -- D0 -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63 --

Where: P PC command: A 1 nibble
 K Sub-command 4 bits
 Bit 3: N/U
 Bit 2-0: 0x5: Bus action
 BB Number of bytes in packet (MAXIMUM 71) 1 byte
 TT Bus address / Serial number 1 byte
 D0 Action to do 1 byte
 00 = Do not use
 01 = Scan
 02 = Remove module
 03 = Reset panel (stops clock to activate watchdog)
 VV Checksum 1 byte

Note: Data byte numbers 16 to 63 are empty because of the bus protocol limitation.

Message: Un / Locate GLCD Keypad.

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...70
TX: PK BB TT -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63 --

Where: P PC command: A 1 nibble
 K Sub-command 4 bits
 Bit 3: N/U
 Bit 2-0: 0x6: Un/Locate Graphic LCD
 BB Number of bytes in packet (MAXIMUM 71) 1 byte
 T Bus address / Serial number 1 byte
 VV Checksum 1 byte

Note: Data byte numbers 16 to 63 are empty because of the bus protocol limitation.

Message: Start Bitmap transfer (starting address / number of bytes)

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...70
TX: PK BB TT UH HL -- DD DD -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63 --

Where: P PC command: A 1 nibble
 K Sub-command 4 bits
 Bit 3: N/U
 Bit 2-0: 0x7: Bitmap transfer
 BB Number of bytes in packet (MAXIMUM 71) 1 byte
 TT Bus address / Serial number 1 byte
 U Starting sector 1 nibble
 H Starting address high / 16 1 byte
 L Starting address low / 16 1 nibble
 DD Number of bytes (0x0001 to 0xFFFF) 2 byte
 VV Checksum 1 byte

Note: The Eeprom Address in this case is always starting from 0x???0.
 Ex: to send data to address 0x12d the starting eeprom address will be 0x0012.

Message: Save Label & broadcast.

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22...70
TX: PS BB -- LT NN NN DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD...-- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15...63 --
```

Where:

P	PC command = B	(1 nibble)
S	Aimed partition (0 to F)	(1 nibble)
	0: Global	
	1: Partition 01	
	2: Partition 02	
	3: Partition 03	
	4: Partition 04	
	5: Partition 05	
	6: Partition 06	
	7: Partition 07	
	8: Partition 08	
	9: Partition 09	
	A: Partition 10	
	B: Partition 11	
	C: Partition 12	
	D: Partition 13	
	E: Partition 14	
	F: Partition 15	
BB	Number of bytes in packet (MAXIMUM 71)	1 byte
L	Message block	(1 nibble)
	0: Characters 001 to 016 or eeprom 0.	
	1: Characters 017 to 032 or eeprom 1 (Mem key).	
	...	
	F: Characters 241 to 256 or eeprom F.	
T	Message Type:	(1 nibble)
	0 = User label	
	1 = Zone label	
	2 = Door label	
	3 = Area Label	
	4 = Module Label	
	5 = Future	
	6 = Future	
	...	
	F = GLCD data (see note)	
N	Message ID (from 0000 to FFFF)	2 bytes
D	LCD message to save	16 bytes
VV	Checksum	1 byte

Note: When the message type is 0x0F, the message number in this case is always starting from 0.
Ex: to send data to message 0x12d, the message number will be 0x0012 and the 13th byte will be loaded with the value. All 16 bytes must be sent to keep existing data.

Message: General Command (Extended 512 bytes of Record data command)

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20...522
TX: P- BH BL SC D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE DF E0...VV
DT: -- -- -- -- 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17...--

Where:

P	PC command = C	(1 nibble)
BH	Number of bytes High (Maximum 524)	1 byte
BL	Number of bytes Low	1 byte
SC	Sub Command:	1 byte
	0 = Reserved	
	1 = Pass-through digiplex	
	2 = Pass-through E-Bus	
	3 = Upload Record	
	4 = Download Record	
	5-FF = Future Use	
D0	Data	519 bytes
..		
VV	Checksum	

Note: This is the general definition of the command. A maximum of 512 bytes of data can be sent as record upload/download.

Message: **General Command Extended (Pass-through Programming)**

NB: 00 01 02 03 04

TX: P- BB BB SC VV

Where:

P	PC command = C	(1 nibble)
BB	Number of byte: 0x0005	2 bytes
SC	Sub Command:	1 byte
	1 = Pass-through digiplex	
	2 = Pass-through E-Bus	
VV	Checksum	

Message: General Command Extended (Upload Records)

NB: 00 01 02 03 04 05 06 07 08 09 10
TX: P- BB BB SC RT RT RN RN NR NR VV
DT: -- -- -- -- 01 02 03 04 05 06 --

Where:

P	PC command = C	(1 nibble)
BB	Number of bytes = 0x000B	2 bytes
SC	Sub Command:	1 byte
	3 = Upload Record	
RT	Record Type High/Low	2 bytes
RN	Record Number High/Low (first record)	2 bytes
NR	Number of Records High/Low	2 bytes
VV	Checksum	1 byte

Message: General Command Extended (Download Records)

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13...521 522
TX: P- BB BB SC RT RT RN RN NR NR DD DD DD DD...DD VV
DT: -- -- -- -- 00 01 02 03 04 05 06 07 08 09...511 --

Where:

P	PC command = C	(1 nibble)
BB	Number of bytes (Maximum 523)	2 bytes
SC	Sub Command:	1 byte
	4 = Download Record	
RT	Record Type High/Low	2 bytes
RN	Record Number High/Low (first record)	2 bytes
NR	Number of Records High/Low	2 bytes
DD	Data in records	
VV	Checksum	1 byte

Message: Unlock System. (NOT USED ANYMORE)

PC:

NB: 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24...37
TX: P- NN NN -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- VV
DT: -- -- -- -- 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20...--

PANEL ANSWER:

NB: 01 02 03 04 05
TX: P- NN NN -- VV
DT: -- -- -- -- 01

Where:	P	Panel Command: C	1 nibble
	N	New safety code	4 nibbles
	VV	Checksum	1 byte

Message: Zone status change.

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22...69 70
TX: P- BB AA CC -- -- D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE DF E0...DD VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63 --

Where:

P	Panel Command: D	1 nibble
BB	Number of bytes in packet (MAXIMUM 71)	1 byte
AA	Action to do on bit	8 bits
CC	Bit to affect	8 bits
D0	Zone 001 to zone 008	8 bits
D1	Zone 009 to zone 016	8 bits
..		
DB	Zone 089 to zone 096	8 bits
..		
E8	Zone 185 to zone 192	8 bits
VV	Checksum	1 byte

Message: request event number xxxx from event buffer.

PC:

NB: 00 01 02 03 04 05 06 07
TX: P- BB -- -- NN NN KK VV
DT: -- -- -- -- -- -- -- --

Where:	P	Panel Command: E	1 nibble
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	NN	Event number (0000 to 2047)	2 bytes
	KK	Number of events requested (1 to 5)	1 byte
	VV	Checksum	1 byte

LIST OF COMMANDS (Panel to PC)

Message: Initialization.

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...36
TX: PO AA -- -- II SS RR CC MM MM WW WW ID E2 EV FW FW NN NN NN NN DD DD DD...VV
DT: -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19...--

Where:	P	PC Command: 0	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	A	Module address (00= panel/module)	1 byte
	I	Product ID	1 byte
	S	Software version	1 byte
	R	Software revision	1 byte
	C	Software ID	1 byte
	M	Module ID	1 byte
	W	Module Password	1 byte
	ID	Winload ID	1 byte
	E2	Memory map Version / Revision	2 nibbles
	EV	Events list Version / Revision	2 nibbles
	FW	Firmware Build number	4 nibbles
	N	Module's serial number	8 nibbles
	D	EVO: Section 3030 to 3038 data	9 bytes
	VV	Checksum	1 byte

Message: Login confirmation.

NB: 00 01 02 03 04 05
TX: PO BB K- CC CC VV For Winload software
TX: PO BB KN NN AA VV For NEware software
DT: -- -- -- -- -- --

Where:	P	PC Command: 1	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	K	Bit 7: Future use	
		Bit 6: Future use	
		Bit 5: Future use	
		Bit 4: 0 = Winload answer	
		1 = NEware answer	
	C	Call back	4 nibble
		0x0000: Call back disabled	
		0xFFFF: Call back enabled	
	N	User number (when access granted)	3 nibbles
	A	User assignment	8 bits
		Bit 7: assigned to partition 8	
		Bit 6: assigned to partition 7	
		Bit 5: assigned to partition 6	
		Bit 4: assigned to partition 5	
		Bit 3: assigned to partition 4	
		Bit 2: assigned to partition 3	
		Bit 1: assigned to partition 2	
		Bit 0: assigned to partition 1	
	VV	Checksum	1 byte

Note: If call back is enabled in the panel, Winload has to hang-up then the panel will call the PC (using the phone number programmed in the panel).

Message: Optimize modem speed (modem only).

NB: 00 01 02 03 04 05
TX: PO BB -- S1 S2 VV
DT: -- -- -- -- -- --

Where:	P	PC Command: 2	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	S1	Modem speed 1	1 byte
	S2	Modem speed 2 (has to be the same as 1)	1 byte
	VV	Checksum	1 byte

Message: Set panel time confirmation.

NB: 00 01 02 03 04 05 06
TX: PO BB -- -- -- -- VV
DT: -- -- -- -- -- -- --

Where:	P	PC Command: 3	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	VV	Checksum	1 byte

Message: Monitoring command confirmation.

NB: 00 01 02 03 04 05 06
TX: PO BB -- -- -- -- VV
DT: -- -- -- -- -- -- --

Where:	P	PC Command: 4	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	VV	Checksum	1 byte

List of commands:	0:	No effect
	1:	Future use
	2:	Full arm partition
	3:	Stay arm partition
	4:	Instant arm partition
	5:	Force arm partition
	6:	Disarm partition
	7:	Future use
	8:	Beep all keypads of partition
	9:	Future use
	A:	Future use
	B:	Future use
	C:	Future use
	D:	Future use
	E:	Future use
	F:	Future use

NB:	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21...69	70
TX:	PO	BB	TT	CC	HH	LL	DD	DD	DD	DD	DD	DD	DD	DD	DD	DD	DD	DD	DD	DD	DD	DD...DD	VV
DT:	--	--	--	--	--	--	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15...63	--

- 49 -

**Message: Download (write) data into module's Eeprom/RAM at address
HHLL confirmation.**

NB: 00 01 02 03 04 05 06
TX: PO BB TT CC HH LL VV
DT: -- -- -- -- -- -- --

Where:	P	PC command: 6	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	TT	Bus address	1 byte
		0x00 = Panel	
		0x01 to 0xFE = Modules	
	CC	Control Byte	8 bits
		Bit 7: 0 = Eeprom access / 1 = RAM access	
		Bit 6: Future use	
		Bit 5: Future use	
		Bit 4: Future use	
		Bit 3: Future use	
		Bit 2: Future use	
		Bit 1: Eeprom address bit 17	
		Bit 0: Eeprom address bit 16	
	HH	Eeprom/RAM Address high	
	LL	Eeprom/RAM Address low	1 byte
	VV	Checksum	1 byte

Message: Error message.

NB: 00 01 02 03 04 05 06
TX: PO BB MM -- -- -- VV
DT: -- -- -- -- -- -- --

Where:	P	Panel Command: 7	1 nibble
	O	Options:	4 bits
		Bit 3:	Future use
		Bit 2:	Alarm reporting pending
		Bit 1:	Winload software is connected
		Bit 0:	NEware software is connected
	BB	Number of bytes in packet	(MAXIMUM 71) 1 byte
	M	Message to Winload:	1 byte
		0x00	:Requested command did not work.
		0x01	:User code is invalid.
		0x02	:Partition in code lockout (too many bad entries).
		0x03	:Future use
		0x04	:Future use
		0x05	:Panel Will Disconnect
		0x06-0x0F	:Future use
		0x10	:Panel Not Connected
		0x11	:Panel Already Connected
		0x12	:Invalid PC Password
		0x13	:Winload on phone line
		0x14	:Invalid Module Address
		0x15	:Cannot Write in RAM
		0x16	:Request to Upgrade Failed
		0x17	:Record Number Out of Range
		0x18	:Reserved
		0x19	:Invalid Record Type
		0x1A	:Multi-Bus Not Supported
		0x1B	:Incorrect number of users
		0x1C	:Invalid Label Number
		0x1D	:Future use
	VV	Checksum	1 byte

Note: Sent to tell PC that software request did not execute correctly.

Message: Save last uploaded event number confirmation.

NB: 00 01 02 03 04 05 06
TX: PO BB -- -- NN NN VV
DT: -- -- -- -- -- -- --

Where:	P	Panel Command: 8	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	N	Event number (16 bit pointer)	4 nibbles
	VV	Checksum	1 byte

Message: Save/read user information.

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...70
TX: PO BB -- NN HH LL D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE D0 -- -- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...64 --

Where:	P	PC command: 9	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
BB		Number of bytes in packet (MAXIMUM 71)	1 byte
NN		Number of user to save or read (up to 4)	1 byte
UH		User number high	1 byte
UL		User number low	1 byte
DD		Data bytes (up to 64 bytes)	64 bytes
VV		Checksum	1 byte

Note : When saving the command has 7 bytes long.

When reading the user information is sent by the panel in data bytes.

Message: General command (general description) confirmation.

NB: 00 01 02 03 04 05 06
TX: PO BB TT CC HH LL VV
DT: -- -- -- -- -- -- --

Where:	P	PC command: A	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	TT	Bus address	1 byte
		0x00 = Panel	
		0x01 to 0xFE = Modules	
	CC	Control Byte	8 bits
		Bit 7: 0 = Eeprom access / 1 = RAM access	
		Bit 6: Future use	
		Bit 5: Future use	
		Bit 4: Future use	
		Bit 3: Future use	
		Bit 2: Future use	
		Bit 1: Eeprom address bit 17	
		Bit 0: Eeprom address bit 16	
	HH	Address high / partition aimed	1 byte/8 bits
	LL	Address low / partition aimed	1 byte/8 bits
	VV	Checksum	1 byte

Message: General command sub command: general broadcast confirmation.

NB: 00 01 02 03 04 05 06
TX: PO BB TT CC HH LL VV
DT: -- -- -- -- -- -- --

Where:	P	PC command: A	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	TT	Bus address	1 byte
		0x00 = Panel	
		0x01 to 0xFE = Modules	
	CC	Control Byte	8 bits
		Bit 7: 0 = Eeprom access / 1 = RAM access	
		Bit 6: Future use	
		Bit 5: Future use	
		Bit 4: Future use	
		Bit 3: Future use Bit 2: Future use	
		Bit 1: Eeprom address bit 17	
		Bit 0: Eeprom address bit 16	
	HH	Address high	1 byte
	LL	Address low	1 byte
	VV	Checksum	1 byte

Message: General command sub command: LCD message OFF confirmation.

NB: 00 01 02 03 04 05 06
TX: PO BB TT CC HH LL VV
DT: -- -- -- -- -- -- --

Where:

P	PC command: A	1 nibble
O	Message center	4 bits
	Bit 3: Future use	
	Bit 2: Alarm reporting pending	
	Bit 1: Winload software is connected	
	Bit 0: NEware software is connected	
BB	Number of bytes in packet (MAXIMUM 71)	1 byte
TT	Bus address	1 byte
	0x00 = Panel	
	0x01 to 0xFE = Modules	
CC	Control Byte	8 bits
	Bit 7: 0 = Eeprom access / 1 = RAM access	
	Bit 6: Future use	
	Bit 5: Future use	
	Bit 4: Future use	
	Bit 3: Future use Bit 2: Future use	
	Bit 1: Eeprom address bit 17	
	Bit 0: Eeprom address bit 16	
HH	partition aimed	
	Bit 0: Partition 1	
	Bit 1: Partition 2	
	Bit 2: Partition 3	
	Bit 3: Partition 4	
	Bit 4: Partition 5	
	Bit 5: Partition 6	
	Bit 6: Partition 7	
	Bit 7: Partition 8	
LL	partition aimed	
	Bit 0: Partition 9	
	Bit 1: Partition 10	
	Bit 2: Partition 11	
	Bit 3: Partition 12	
	Bit 4: Partition 13	
	Bit 5: Partition 14	
	Bit 6: Partition 15	
	Bit 7: Future use	
VV	Checksum	1 byte

Message: **General command sub command: LCD message LOW priority confirmation.**

NB: 00 01 02 03 04 05 06
TX: PO BB TT CC HH LL VV
DT: -- -- -- -- -- -- --

Where:	P	PC command: A	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	TT	Bus address	1 byte
		0x00 = Panel	
		0x01 to 0xFE = Modules	
	CC	Control Byte	8 bits
		Bit 7: 0 = Eeprom access / 1 = RAM access	
		Bit 6: Future use	
		Bit 5: Future use	
		Bit 4: Future use	
		Bit 3: Future use	
		Bit 2: Future use	
		Bit 1: Eeprom address bit 17	
		Bit 0: Eeprom address bit 16	
	HH	partition aimed	
		Bit 0: Partition 1	
		Bit 1: Partition 2	
		Bit 2: Partition 3	
		Bit 3: Partition 4	
		Bit 4: Partition 5	
		Bit 5: Partition 6	
		Bit 6: Partition 7	
		Bit 7: Partition 8	
	LL	partition aimed	
		Bit 0: Partition 9	
		Bit 1: Partition 10	
		Bit 2: Partition 11	
		Bit 3: Partition 12	
		Bit 4: Partition 13	
		Bit 5: Partition 14	
		Bit 6: Partition 15	
		Bit 7: Future use	
	VV	Checksum	1 byte

Message: General command sub command: LCD message HIGH priority confirmation.

NB: 00 01 02 03 04 05 06
TX: PO BB TT CC HH LL VV
DT: -- -- -- -- -- -- --

Where:

P	PC command: A	1 nibble
O	Message center	4 bits
	Bit 3: Future use	
	Bit 2: Alarm reporting pending	
	Bit 1: Winload software is connected	
	Bit 0: NEware software is connected	
BB	Number of bytes in packet (MAXIMUM 71)	1 byte
TT	Bus address	1 byte
	0x00 = Panel	
	0x01 to 0xFE = Modules	
CC	Control Byte	8 bits
	Bit 7: 0 = Eeprom access / 1 = RAM access	
	Bit 6: Future use	
	Bit 5: Future use	
	Bit 4: Future use	
	Bit 3: Future use Bit 2: Future use	
	Bit 1: Eeprom address bit 17	
	Bit 0: Eeprom address bit 16	
HH	partition aimed	
	Bit 0: Partition 1	
	Bit 1: Partition 2	
	Bit 2: Partition 3	
	Bit 3: Partition 4	
	Bit 4: Partition 5	
	Bit 5: Partition 6	
	Bit 6: Partition 7	
	Bit 7: Partition 8	
LL	partition aimed	
	Bit 0: Partition 9	
	Bit 1: Partition 10	
	Bit 2: Partition 11	
	Bit 3: Partition 12	
	Bit 4: Partition 13	
	Bit 5: Partition 14	
	Bit 6: Partition 15	
	Bit 7: Future use	
VV	Checksum	1 byte

Message: General command sub command: PGM override confirmation.

NB: 00 01 02 03 04 05 06
TX: PO BB TT CC HH LL VV
DT: -- -- -- -- -- -- --

Where:	P	PC command: A	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	TT	Bus address	1 byte
		0x00 = Panel	
		0x01 to 0xFE = Modules	
	CC	Control Byte	8 bits
		Bit 7: 0 = Eeprom access / 1 = RAM access	
		Bit 6: Future use	
		Bit 5: Future use	
		Bit 4: Future use	
		Bit 3: Future use Bit 2: Future use	
		Bit 1: Eeprom address bit 17	
		Bit 0: Eeprom address bit 16	
	VV	Checksum	1 byte

Message: Bus action confirmation.

NB: 00 01 02 03 04 05 06
TX: PO BB TT CC HH LL VV
DT: -- -- -- -- -- -- --

Where:	P	PC command: A	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	TT	Bus address	1 byte
		0x00 = Panel	
		0x01 to 0xFE = Modules	
	CC	Control Byte	8 bits
		Bit 7: 0 = Eeprom access / 1 = RAM access	
		Bit 6: Future use	
		Bit 5: Future use	
		Bit 4: Future use	
		Bit 3: Future use Bit 2: Future use	
		Bit 1: Eeprom address bit 17	
		Bit 0: Eeprom address bit 16	
	VV	Checksum	1 byte

Message: Label broadcast confirmation / Data From Bus Module (serial Only).

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21...69 70
TX: PO BB -- LT NN NN DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD...-- VV
DT: -- -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15...63 --
```

Where:	P	PC command: B	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	L	Message block	1 nibble
		0: Characters 001 to 016 or eeprom 0.	
		1: Characters 017 to 032 or eeprom 1 (Mem key).	
		...	
		F: Characters 241 to 256 or eeprom F.	
	T	Message Type:	1 nibble
		0 = User label	
		1 = Zone label	
		2 = Door label	
		3 = Area Label	
		4 = Module label	
		5 = Future	
		6 = Future	
		...	
		E = Data to broadcast to serial port	
		F = GLCD data (see note)	
	N	Message ID (from 0000 to FFFF)	2 bytes
	DD	LCD message to save	up to 16 bytes
	VV	Checksum	1 byte

Note: When the message type is 0x0F, the message number in this case is always starting from 0.
Ex: to send data to message 0x12d, the message number will be 0x0012 and the 13th byte will be loaded with the value. All 16 bytes must be sent to keep existing data.

Message: Special Data From Bus Module (serial Connection Only).

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21...69 70
TX: PO BB -T AA MM -- DD...-- VV
DT: -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15...63 --

Where:	P	PC command: B	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	T	Message Type = 0xE	1 nibble
	AA	Module's Bus address	1 byte
	MM	Message ID (from 00 to FF)	1 byte
	DD	Data sent from module	up to 16 bytes
	VV	Checksum	1 byte

Note: This Command is sent from a Module on the Bus to the software connected to the serial port.

Message: General Command Panel Answer(Extended 512 bytes of Record data command)

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20...522
TX: P- BH BL SC D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE DF E0...VV
DT: -- -- -- -- 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17...--

Where:

P	PC command = C	(1 nibble)
BH	Number of bytes High (Maximum 523)	1 byte
BL	Number of bytes Low	1 byte
SC	Sub Command:	1 byte
	0 = Reserved	
	1 = Pass-through digiplex	
	2 = Pass-through E-Bus	
	3 = Upload Record	
	4 = Download Record	
	5-FF = Future Use	
D0	Data	519 bytes
..		
VV	Checksum	

Note: This is the general definition of the command. A maximum of 512 bytes of data can be sent as record upload/download.

Message: General Command Extended (Pass-through Programming Request Result)

NB: 00 01 02 03 04 05
TX: P- BB BB SC RR VV

Where:

P	PC command = C	(1 nibble)
BB	Number of byte: 0x0006	2 bytes
SC	Sub Command:	1 byte
	1 = Pass-through digiplex	
	2 = Pass-through E-Bus	
RR	Result Code:	1 byte
	0x00 :Command successfully executed	
	0x85 :System in programming	
	0x88 :Partition in Alarm	
	0x89 :Partition in Entry Delay	
	0x8A :Partition Armed	
	0x8C :Partition in Exit Delay	
VV	Checksum	

Message: General Command Extended (Upload Records)

NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13...521 522
TX: P- BB BB SC RT RT RN RN NR NR DD DD DD DD...DD VV
DT: -- -- -- -- 00 01 02 03 04 05 06 07 08 09...511 --

Where:

P	PC command = C	(1 nibble)
BB	Number of bytes	2 bytes
SC	Sub Command:	1 byte
	3 = Upload Record	
RT	Record Type High/Low	2 bytes
RN	Record Number High/Low (first record)	2 bytes
NR	Number of Records High/Low	2 bytes
DD	Data in records	
VV	Checksum	1 byte

Message: General Command Extended (Download Records)

NB: 00 01 02 03 04 05 06 07 08 09 10
TX: P- BB BB SC RT RT RN RN NR NR VV
DT: -- -- -- -- 01 02 03 04 05 06 --

Where:

P	PC command = C	(1 nibble)
BB	Number of bytes = 0x000B	2 bytes
SC	Sub Command:	1 byte
	4 = Download Record	
RT	Record Type High/Low	2 bytes
RN	Record Number High/Low (first record)	2 bytes
NR	Number of Records High/Low	2 bytes
VV	Checksum	1 byte

Message: Unlock System confirmation. (not used anymore)

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23...37
TX: PO NN NN -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- VV
DT: -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19...--
```

```
NB: 00 01 02 03 04
TX: PO NN NN -- VV
DT: -- -- -- -- 00
```

```
Where:      P      Panel Command: C                      1 nibble
            O      Message center                        4 bits
            Bit 3:   Future use
            Bit 2:   Alarm reporting pending
            Bit 1:   Winload software is connected
            Bit 0:   NEware software is connected
            N      New safety code (4 Digit)              4 nibbles
            V      Checksum                              1 byte
```

Message: Zone status change confirmation.

NB: 00 01 02 03 04 05 06
TX: PO BB AA CC -- -- VV
DT: -- -- -- -- -- -- --

Where:	P	Panel Command: D	1 nibble
	O	Message center	4 bits
		Bit 3: Future use	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	A	Action to do on bit	8 bits
	B	Bit to affect	8 bits
	V	Checksum	1 byte

Message: Event transmission.

ANSWER TO PC:

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22...69 70
TX: PO BB -- SS NN NN E1 E1 E1 E1 E1 E1 E1 E1 E1 E1 E1 E2 E2 E2 E2 E2...EX VV
DT: -- -- -- -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...63 --
```

LIVE EVENTS:

```
NB: 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20...34 35 36
TX: PO SS CC CC D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE LT L0 L1...LF -- VV
DT: -- -- -- -- 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16...-- -- --
```

Where:	P	Panel Command: E	1 nibble
	O	Message center	4 bits
		Bit 3: IP100 Report enable(0)/disable(1)	
		Bit 2: Alarm reporting pending	
		Bit 1: Winload software is connected	
		Bit 0: NEware software is connected	
	BB	Number of bytes in packet (MAXIMUM 71)	1 byte
	SS	0x00: Requested event number	
		0xFF: Event just occurred (serial port only)	
	CC	Live Event Counter (0000 to 2047)	2 bytes
	NN	Event number (0000 to 2047)	2 bytes
	D1	Century	1 byte
	D2	Year	1 byte
	D3	Month	1 byte
	D4	Day	1 byte
	D5	Hour	1 byte
	D6	Minute	1 byte
	D7	Event group (see list below)	1 byte
	D8	Number 1 or 2 high nibble / Partition	2 nibbles
	D9	Event Number 1	2/3 nibbles
	DA	Event Number 2	2/3 nibbles
	DB	Serial number 1 & 2	1 byte
	DC	Serial number 3 & 4	1 byte
	DD	Serial number 5 & 6	1 byte
	DE	Serial number 7 & 8	1 byte
	E1	Event 1 requested (in compressed format)	12 bytes
	E2	Event 2 requested (in compressed format)	12 bytes
	LT	Label Type:	1 byte
		0 = No Label	
		1 = User label	
		2 = Zone label	
		3 = Door label	
		4 = Area Label	
		5 = Module label	
		6 = Future	
		..	
	Lx	Label	16 bytes
	V	Checksum	1 byte