Domintell LightProtocol guide for communication interface of new generatation (DGQG02,...).

The goal of this document is to describe Domintell's Ethernet interfaces of new generation (DGQG02, DGQG03, DGQG04). It will help you to make the good choice between the options available. Input specifications are the same for all modules (data to Domintell). Output protocol specifications are different (data from Domintell)

DGQG02, DGQG03, DGQG04 :
Functionalities depend on the firmware. These modules are explained below in details.

For informations about DRS3202, DETH02 or any older LightProtocol interfaces, please consider reading DS_RS232_ETH_Interfaces_v1_27_08.pdf

Table of Contents

1. Document revisions

v5 : 27/10/2020
Fix typo in description of "TRP 151-4"

v4 : 21/09/2020
Fix typo for DGQG02 commands with data (command and data must be seperated with '|')
Add format of statuses for TypeSensorIo, TypeMovIo, TypeLuxIo, TypeHumidityIo, TypePressureIo, TypeCo2Io
Add example for NewGen statuses

v3 : 26/08/2020
Add DTSC05, DDMX02
Add example for NewGen commands
Add minimal DAP version required to handle modules
Adjust header of APPINFO block

v2 : 24/10/2018
Initial release

2. Informations about communication interfaces

2.1. General information

The goal of this document is to describe Ethernet interfaces of new generation for Domintell2 system and to help you to make the good choice between the options available. The hardware does not change but the functions depend on the firmware.
There is several ways to communicate with Domintell system depending of the module (some of these modules are deprecated. There are just named for information):

- Custom input ASCII strings (sent to Domintell system). need creation of links in configuration software (See chapter "Parameters and specific links->DRS23201 module" in Domintell2 Configuration software manual) is working with DRS23201, DRS23202, DETH02, DUSB01 and DGSM01.
- Custom output ASCII strings (sent to your device). need creation of links in configuration software (See chapter "Parameters and specific links->DRS23201 module" in Domintell2 Configuration software manual) is working with DRS23201, DUSB01 and DGSM01.
- Legacy output LightProtocol (LightProtocol server to LightProtocol client) is only available on DNET01, DGQG02, DGQG03, DGQG04, DRS23202 and DETH02. No configuration/link is required in Domintell2 configuration software; it is automatically generated by master module.
- Legacy Input LightProtocol (LightProtocol client to LightProtocol server) is working with DNET01, DGQG02, DGQG03, DGQG04, DRS23201, DRS23202, DETH02, DUSB01 and DGSM01. No configuration/link is required in Domintell2 configuration software; it is automatically decoded by master module.

This document only covers Ethernet modules of new generation (like DNET01, DGQG02, …) using Secured Websockets instead of RS232 or UDP socket.

2.2. Devices overview

Here is the list of communication modules and their capabilities :

- DNET01 :
  - IP : DHCP or static. It is highly recommended to set a static IP.
  - Default port 17481.
  - Possibility to set a password.
  - Limited to max 1 legacy UDP (almost backward compatible for application written for DETH02)
  - Limited to max 8 simultaneous connections using Secured WebSocket protocol.
  - Legacy Input LightProtocol (LightProtocol client to LightProtocol server);
  - Custom input ASCII strings (sent to Domintell system).;
  - Legacy output LightProtocol (LightProtocol server to LightProtocol client).

- DGQG02/DGQG03/DGQG04 :
  - Limited to max 2 simultaneous connections using Secured WebSocket protocol.

- Legacy Input LightProtocol (LightProtocol client to LightProtocol server);
- Custom input ASCII strings (sent to Domintell system).;
- Legacy output LightProtocol (LightProtocol server to LightProtocol client).

2.3. Ethernet wiring information

The RJ45 connector must be connected to the LAN (Local Area Network) with a classic UTP RJ45 Cable (CAT5) to a switch or a router.

WARNING :
Do NOT connect Domintell bus on the RJ45 connector, this can cause fatal damages to the module.

3. Terminology

- Master : Refers to DGQG0x modules.
- Central Unit : Refers to DGQG0x modules.
- Third-party device : device that needs to control Domintell2 installation. This can be a smartphone, a computer, a Raspberry Pi, ...
- LightProtocol server : refer to module on Domintell system that generates APPINFO and statuses. DGQG0x, DNET01 are LightProtocol servers.
- LightProtocol client : refer to third-party application/device that will interact with Domintell system by parsing APPINFO sent by LightProtocol server and sending command to LightProtocol server to control outputs. Smartphone apps like DomintellPilot is a LightProtocol client.
- Input frame : frame sent from LightProtocol client to LightProtocol server

4. Protocol specifications

4.1. Abreviation of Modules' type

Reference	Mod Type	Min. DAP version	Description	Possible output data type
DAMPLI01	AMP	30	Sound Module	S
DBIR01	BIR	30	8 bipolar relays	O
DDIM01	DIM	30	8 dimmer commands	D
DDIR01	DIR	30	IR detector	C
DMV01	DMV	30	Mechanical ventilation	O
DDMX01	DMX	30	DMX Module	X
DDMX02	DX2	33	DMX Module with RGBW handling	Please see 4.5.j IO type list and data format
DENV01	EV1	33	Environment sensor module	Please see 4.5.j IO type list and data format
DENV02	EV2	33	Environment sensor module with CO2 sensor	Please see 4.5.j IO type list and data format
DFAN01	FAN	30	Fan controler	O for security reasons, valves always follow the setpoint regulation, so if you need to toggle the valves of the DFAN01, you must first change the setpoint on the associated sensor. If valves are OFF, fan will not start. 6th DFAN01 ouput is the working mode : 0 = auto, 1 = manual.
DGQG02	QG2	30	New generation Master module with embedded IOs	*Please see 4.5.j IO type list and data format*

Reference	Mod Type	Min. DAP version	Description	Possible output data type
DGQG03	QG3	n/a	New generation Master module with embedded IOs and DALI® interface	*Please see 4.5.j IO type list and data format*
DGQG04	QG4	30	New generation Master that replaces DGQG01	*None*
DIN10V01	I10	30	Analog 0-10V input module	D
DINTDALI01	DAL	30	DALI interface	D
DISM04	IS4	30	4 Inputs module	I
DISM08	IS8	30	8 Inputs module	I
DISM20	I20	30	20 inputes module	I
DLED01	LED	30	4 leds driver	O
DMOV01 DMOV02 DMOV05	DET	30	Infrared detector	I
DMOV06	MV6	33	Motion and light detector	Please see 4.5.j IO type list and data format
DMR01	DMR	30	5 Monopolar relays	O
DOUT10V02	D10	30	0/1-10V dimmer module	D
DPBC01	CL1	30	1 Push Button Classic (8 colors and temperature sensor)	I,O,T
DPBC02	CL2	30	2 Push Button Classic (8 colors and temperature sensor)	I,O,T
DPBC04	CL4	30	4 Push Button Classic (8 colors and temperature sensor)	I,O,T
DPBC06	CL6	30	6 Push Button Classic (8 colors and temperature sensor)	I,O,T
DPBL01	B81	30	1 Push Button Lythos (and 8 colors)	I,O
DPBL02	B82	30	2 Push Button Lythos (and 8 colors)	I,O
DPBL04	B84	30	4 Push Button Lythos (and 8 colors)	I,O
DPBL06	B86	30	6 Push Button Lythos (and 8 colors)	I,O
DPBR02	BR2	30	2 Push Button Rainbow (and	I,O

Reference	Mod Type	Min. DAP version	Description	Possible output data type
			RGB)	
DPBR04	BR4	30	4 Push Button Rainbow (and RGB)	I,O
DPBR06	BR6	30	6 Push Button Rainbow (and RGB)	I,O
DPB(U/T)01	BU1	30	1 Push Button	I,O
DPB(U/T)02	BU2	30	2 Push Button	I,O
DPB(U/T)04	BU4	30	4 Push Button	I,O
DPB(U/T)06	BU6	30	6 Push Button	I,O
DPBRLCD02	PRL	30	Rainbow LCD push buttons	B,O,T,U,M,R
DPBTLCD0x	PBL	30	LCD push buttons	B,O,T,U,M,R (T,U,M,R = DPBTLCD02 only)
DTEM01	TE1	30	Temperature sensor	T,U,M,R
DTRP01	TRP	30	4 teleruptors	O
DTRP02	TPV	30	2 shutter command with teleruptors Bit 0 Relay 1 = UP Bit 1 Relay 1 = DOWN ...	O
DTRV01	TRV	30	4 shutter inverters Bit 0 Relay 1 = UP Bit 1 Relay 1 = DOWN ...	O
DTRVBT01	V24	30	1 DC shutter command Bit 0 = UP – Bit 1 = DOWN	O (Low voltage TRV – 1 out – available soon)
DTSC02	LT2	30	TFT Touchscreen	I,T,U,M,R
DTSC04	LT4	30	TFT Touchscreen with video	I,T,U,M,R
DTSC05	LT5	33	Rainbow capacitive Touchscreen	Please see 4.5.j IO type list and data format
DVIP01	VI1	30 only	1 button videophone	I
DVIP02	VI2	30 only	2 buttons videophone	I
DWIND01	WI1	33	Wind sensor module	
ModBus Device	MBD	30	Ex: Daikin RTD-NET	T,U,M,R,D
Cameras	CAM	30	Cameras informations	
Clocks	CLK	30	Programmes clock (normal, reset and astronomical)	K

Reference	Mod Type	Min. DAP version	Description	Possible output data type
Radio Station	STA	30	Radio Station name & frequency	
Software Vars	VAR	30	Virtual programmed status	O,D,M,R (serial = number in order of appearance on the configuration screen) So you'll be able to create different events.
System Vars	SYS	30	See 4.2 Information about system variables	O (Since v1.12.01 & higher), 9
Temp. Plage List	TPL	30	Specific range of a Temp. profile	P
Temp. Profile	TPR	30	Profile's name which contains next Temp. plage lists received	

Min. DAP version is given by the following string in APPINFO:
`APPINFO (PROG M 33.0 00/00/00 00h00 Rev=0) => TEST_APPINFO.dap`

4.2. Information about system variables

Code	Name	Data type	Value
SYS000000	Simulation mode	Bool 'o'	0 = Simulation is not playing (only record) 1 = Simulation is playing
SYS000009	Daytime	Bool 'o'	Based on astronomical clock 0 = Night 1 = Daytime

4.3. Legacy Input LightProtocol (*LightProtocol client* to *LightProtocol server*)

4.3.a) Overview

These commands/strings can be sent to Domintell2 system and are executed without doing any links (Automatic LightProtocol).
Please read 4.5.b Modules using new generation input LightProtocol to know which frame format must be used for a given module.

4.3.b) General recommandations/limitations

- Frames **can not** be concatenated anymore using '&'.
- Important : we advise you to make less than 100 «string» links on the same intput because it's a lot of work for the Master (DGQG0x). A WARNING will be displayed into the Diagnose function if there's more than 100 «string» links.
- LightProtocol strings have priority on ASCII (custom) string. If a link is done in Domintell2 configuration software using text "BIR000B4B-1", master unit will decode it as LightProtocol string and will not execute your link.
- Strings '<CR>', '<LF>' and '<TAB>' are replaced by the equivalent ASCII code : 0x0D, 0x0A and 0x09.
- Carriage return & line feed characters are supported at the end of the command line.
- Domintell Automatically suppress (trim) the SPACE characters at the begin or at the end of the message.
- Strings are NOT case sensitive. Lower case characters are automatically replaced with upper case equivalent. (Be careful with accentuated characters like "éèêàñäí"...)
- We advise to use only ASCII characters. Accentuated character can be coded over multiple bytes under UTF-8 systems.

4.3.c) Legacy input frame format

Mod Type (3 char)	Serial Number (6 char hexadecimal)	- (1 char)	Output Number (1 char)	Action parameters

4.3.d) Legacy action parameters

A parameter always start with the character '%' (reserved char)

Char	**Means**	**Description** (leading '0' can be replaced by ' ' (space))
`I`	Inputs	LSB = input 0, MSB = input 7
`O`	Outputs	LSB = output 0, MSB = output 7
`D`	Dimmers	2 bytes by output (%) Example : '`64`' = 100%
`X`	DMX	2 bytes by channel Example : '`C0`' = 192
`T`	Temperature Heating setpoint	Example : '`20.5 22.0 AUTO 18.0`' 1st T° = measure (with software offset) 2nd T° = Heating setpoint value Sensor T° Mode

Char	Means	Description (leading '0' can be replaced by ' ' (space))
		3rd T° = Heating profile value
U	Temperature Cooling setpoint	Example : '20.5 22.0 HEATING 18.0' 1st T° = measure (with software offset) 2nd T° = cooling setpoint value Sensor Regulation Mode 3rd T° = cooling profile value
C	Infrared Command	Example : Key 1 = '01'
S	Sound	'1-32-TUNE-63-03E8' = Output 1 – 50% - Source Tuner – 99,1000 Mhz (Since card version 5)
B	Button	2 bytes(button number) + 2 bytes (00=released 01=pressed)
P	Temp. Plage	Example : 12:32:00 21.6 1st = hh:mm:ss 2nd = setpoint value
K	Clocks	Example : 00:38:00 7F 00/01/04 Clock 1st = hh:mm:ss 2nd = Day mask (b0=sunday, b1=monday, ... b7= disable clock (=1)) 3rd = Name 4th = Type of clock : blank (normal), SUNSET, SUNRISE, RESET

- '%Dxxx' decimal dimmer/volume value assignment
- '%DB' and '%DE' : execute a Start/Stop dim on a dimmer output
- '%I%Dxxx' and '%O%Dxxx' Increase and Decrease dimmer/volume value by step of decimal 'xxx' percent
- '%Txx.x' decimal T° value (set Heating setpoint)
- '%Uxx.x' decimal T° value (set Cooling setpoint)
- '%Ax' Sound Auxiliary selection 1=>4, Tuner = 5
- '%Fxxx,xxxx'decimal Tuner Frequency in Mhz
- '%I' set the output
- '%O' reset the output
- '%Mx' set Temperature mode (1=absence, 2=auto, 5=confort, 6=gel)
- '%Rx' set Regulation mode (0=off, 1=heating, 2=cooling, 3=mixed)
- '%H' shutter goes High
- '%L' shutter goes Low
- '%S' ask status of module (does not work with MEMO)
- '%Px' simulate a push on an input (1=Begin short push, 2=End short push, 3=Begin long push, 4=End long push)

4.3.e) Samples of legacy strings sent to your Domintell installation

Text	Means
BU1 11-1	Change/toggle output/led 1 on module DPBU01 with serial number 0x000011

`BU1 11-1%I`	Set (ON) output/led 1 on module DPBU01 with serial number 0x000011
`BU1 11-1%0`	Reset (OFF) output/led 1 on module DPBU01 with serial number 0x000011
`BU1 11%S`	Get Status of input (button) and output (LED's) on module DPBU01 with serial number 0x000011
`BU2 52-2`	Change/toggle output/led 2 on module DPBU02 with serial number 0x000052
`BU4 4F-4`	Change/toggle output 4 on module DPBU04 with serial number 0x00004F and
`BU2 52-2%P1`	Simulate Begin of short push on button 2 of module DPBU02 with serial number 0x000052
`BU6 134-1%P2`	Simulate End of short push on button 1 of module DPBU06 with serial number 0x000134
`IS4 CD-4%P3`	Simulate Begin of long push on input 4 of module DISM04 with serial number 0x0000CD
`IS8 2D8-7%P4`	Simulate End of long push on input 7 of module DISM08 with serial number 0x0002D8
`BIR 3A6-8`	Change output 8 on module DBIR01 with serial number 0x0003A6
`TRV 73-1 or` `TRV 73-2`	Run shutter function of shutter 1 on module DTRV01 with serial number 0x000073. Each time the command is send one step of the following cycle is run : UP-STOP-DOWN-STOP-UP-…

`TRV 73-1%H or`
`TRV 73-1%I or`
`TRV 73-2%H`
	Shutter 1 on module DTRV01 with serial number 0x000073 goes High.
	Returned status will be "TRV 73O01" if not other shutter is ON.
	(since v1.19.17)

`TRV 73-1%L or`
`TRV 73-2%L or`
`TRV 73-2%I`
	Shutter 1 on module DTRV01 with serial number 0x000073 goes Low.
	Returned status will be "TRV 73O02" if not other shutter is ON.
	(since v1.19.17)

`TRV 73-1%O or` `TRV 73-2%O`	Stop shutter 1 on module DTRV01 with serial number 0x000073
`TRV 73-3 or` `TRV 73-4`	Run shutter function of shutter 2 on module DTRV01 with serial number 0x000073. Each time the command is send one step of the following cycle is run : UP-STOP-DOWN-STOP-UP-…

`TRV 73-3%H or`
`TRV 73-3%I or`
`TRV 73-4%H`
	Shutter 2 on module DTRV01 with serial number 0x000073 goes High.
	Returned status will be "TRV 73O04" if not other shutter is ON.
	(since v1.19.17)

`TRV 73-3%L or`
`TRV 73-4%L or`
`TRV 73-4%I`
	Shutter 2 on module DTRV01 with serial number 0x000073 goes Low.
	Returned status will be "TRV 73O08" if not other shutter is ON.
	(since v1.19.17)

`TRV 73-3%O or` `TRV 73-4%O`	Stop shutter 2 on module DTRV01 with serial number 0x000073
`TRV 73-5 or` `TRV 73-6`	Run shutter function of shutter 3 on module DTRV01 with serial number 0x000073. Each time the command is send one step of the following cycle is run : UP-STOP-DOWN-STOP-UP-…

`TRV 73-5%H or`
`TRV 73-5%I or`
`TRV 73-6%H`
	Shutter 3 on module DTRV01 with serial number 0x000073 goes High.
	Returned status will be "TRV 73O10" if not other shutter is ON.
	(since v1.19.17)

`TRV 73-5%L or`
`TRV 73-6%L or`
`TRV 73-6%I`
	Shutter 3 on module DTRV01 with serial number 0x000073 goes Low.
	Returned status will be "TRV 73O20" if not other shutter is ON.
	(since v1.19.17)

`TRV 73-5%O or` `TRV 73-6%O`	Stop shutter 3 on module DTRV01 with serial number 0x000073
`TRV 73-7 or` `TRV 73-8`	Run shutter function of shutter 4 on module DTRV01 with serial number 0x000073. Each time the command is send one step of the following cycle is run : UP-STOP-DOWN-STOP-UP-…

`TRV 73-7%H or`
`TRV 73-7%I or`
`TRV 73-8%H`
	Shutter 4 on module DTRV01 with serial number 0x000073 goes High.
	Returned status will be "TRV 73O40" if not other shutter is ON.
	(since v1.19.17)

`TRV 73-7%L or`
`TRV 73-8%L or`
`TRV 73-8%I`
	Shutter 4 on module DTRV01 with serial number 0x000073 goes Low.
	Returned status will be "TRV 73O80" if not other shutter is ON.
	(since v1.19.17)

`TRV 73-7%O or` `TRV 73-8%O`	Stop shutter 4 on module DTRV01 with serial number 0x000073
`TRP 151-4`	Change output 4 on module DTRP01 with serial number 0x000151
`DIM 19F-8`	Change output 8 on module DDIM01 with serial number 0x00019F
`DIM 19F-6%D50`	Set output 6 to 50% on module DDIM01 with serial number 0x00019F
`DIM 19F-6%DB`	Start dimming on output 6 on module DDIM01

	with serial number 0x00019F (v1.17.02)
DIM 19F-6%DE	Stop dimming on output 6 on module DDIM01 with serial number 0x00019F (v1.17.02)
DIM 19F-6%I%D10	Increase by step of 10% the value on output 6 on module DDIM01 with serial number 0x00019F (stop at 100%) (v1.17.02)
DIM 19F-6%O%D7	Decrease by step of 7% the value on output 6 on module DDIM01 with serial number 0x00019F (stop at 0%) (v1.17.02)
LED C2-1	Change output 1 on module DLED01 with serial number 0x0000C2
VAR 1	Change variable 1
TPV 3-1	Change shutter 1 on module DTRP02 with serial number 0x000003
D10 1-1	Change output 1 on module DOUT10V02 with serial number 0x000001
D10 1-1%D60	Set output 1 to 60% on module DOUT10V02 with serial number 0x000001
D10 1-1%I%D5	Increase output value of module DOUT10V02 with serial number 0x000001 by step of 5% (v1.17.02)
D10 1-1%O%D11	Decrease output value of module DOUT10V02 with serial number 0x000001 by step of 11% (v1.17.02)
DMX 1F-2-1%X230	Set channel 1 of device 2 to value 230 of module DDMX01 with serial number 0x00001F
V24 1-1	Change shutter 1 on module DTRVBT01 with serial number 0x000001
TSB 8D%T24.5	Set Heating T° to 24,5°C on module DTSC01/03 with serial number 0x00008D
LT2 34%T22.7	Set Heating T° to 22,7°C on module DTSC02 with serial number 0x000034
LT4 2F%U21.5	Set Cooling T° to 21,5°C on module DTSC04 with serial number 0x00002F
I10 5%S	Ask Status of the input of DIN10V with serial number 0x000005
AMP 3-1%D50%A1	Output 1 to Aux 1 at Volume 50 on module DAMPLI01 with serial number 0x000003
AMP 3-1%I%D15	Increase volume of Output 1 by step of 15% on module DAMPLI01 with serial number 0x000003 (v1.17.02)
AMP 3-1%O%D9	Decrease volume of Output 1 by step of 9% on module DAMPLI01 with serial number 0x000003

	(v1.17.02)
AMP 3-2%D60%F99.1%A5	Output 2 to Tuner at Volume 60 & Freq 99,1MHz on module DAMPLI01 with serial number 0x000003
AMP000003-4	Change output 4 volume on module DAMPLI01 with serial number 0x000003
AMP000003%S	Ask status of all output of module DAMPLI01 with serial number 0x000003
BIR 3A6-6%I	Set output 6 on module DBIR01 with serial number 0x0003A6
BIR 3A6-6%O	Reset output 6 on module DBIR01 with serial number 0x0003A6
MEM000001%I	SET Mixed Memo 1 (v1.16.02)
MEM000001%O	RESET Mixed Memo 1 (v1.16.02)
MEM000002%D50	SET 50% to Dimmer Memo 2 (v1.16.03)
MEM000002%I%D5	Increase value of Dimmer Memo 2 by step of 5% (v1.17.02)
MEM000002%O%D17	Decrease value of Dimmer Memo 2 by step of 17% (v1.17.02)
MEM 3%O	Shutter Memo Group : OFF
MEM 3%H	Shutter Memo Group : UP (High)
MEM 3%L	Shutter Memo Group : Down (Low))
SFE000001	SET Sfeer 1 (v1.16.03)
SFE000001%I	SET Sfeer 1 (v1.16.03)
SFE000001%S	Get status of each item in the Sfeer 1 (v1.17.02)
PBL C-6%I	SET DPBTLCD0x 6th output
PBL C-1%O	RESET DPBTLCD0x 1st output
PBL C-1%P2	Simulate begin of short push on button 1 of module DPBTLCD0x with serial number 0x00000C (v1.17.02)
PBL 13%S	Return status (Temp -> only for DPBTLCD02) of module DPBTLCD02 with serial number 0x000013 (v1.17.02)
FAN000001-1%I	Set speed 1
FAN000001-2%I	Set speed 2
FAN000001-3%I	Set speed 3
FAN000001-4%I	Set Heating (if speed different of 0) Advise : change T° sensor setpoint!
FAN000001-5%I	Set Cooling (if speed different of 0) Advise : change T° sensor setpoint!

FAN000001-6%I	Set Manual mode
FAN000001-6%O	Set Automatic mode
DMV00001-1%I	Set speed 1
DMV00001-2%I	Set speed 2
DMV00001-3%I	Set speed 3
DMV00001-4%I	Set Auxiliary 1
DMV00001-5%I	Set Auxiliary 2
ZON000001%I	T° Zone 1, increment setpoint. (T° zones since v1.17.00)
ZON000001%O	T° Zone 1, decrement setpoint.
ZON000001%T15.5	T° Zone 1, setpoint to 15.5°C.
ZON000001%M1	T° Zone 1, set T° mode to absence.
ZON000001%M2	T° Zone 1, set T° mode to automatic.
ZON000001%M5	T° Zone 1, set T° mode to comfort.
ZON000001%M6	T° Zone 1, set T° mode to frost (if frost mode enabled).
CLK000001%K00:22:00 7F 00/05/09	Set Clock 1 at 00h22m00s for all weekdays during month of may (v1.17.02)
CLK000001%K00:22:00 FF 00/05/09	Disable Clock 1 and set datas to 00h22m00s for all weekdays during month of may (v1.17.02)
CLK000001%K01:22:00 08 00/00/00	Set Clock 1 at 01h22m00s each Wednesday (v1.17.02)
DAL 10-32%D100	DINTDALI01 #0x10 output 0x32 request @ 100%
PRL C-6%I	SET DPBRLCD02 6th output
PRL C-1%O	RESET DPBRLCD02 1st output
PRL C-1%P2	Simulate begin of short push on button 1 of module DPBRLCD02 with serial number 0x00000C (v1.27.01)
PRL 13%S	Return status of module DPBRLCD02 with serial number 0x000013 (v1.27.01)

4.4. Legacy output LightProtocol (*LightProtocol server* to *LightProtocol client*)

4.4.a) Status Frame description

Mod Type (3 char)	Serial Number (6 char hexadecimal)	(optional) IO number (-x : minus char + IO number in 1 hexa digit) DINTDALI requires 2 hexa digit	Data Type (1 char)	Datas (n * 2 char hexa)

4.4.b) Data Types

Type	***Means***	***Description*** (leading '0' can be replaced by ' ' (space))
`I`	Inputs	LSB = input 0, MSB = input 7
`O`	Outputs	LSB = output 0, MSB = output 7
`D`	Dimmers	2 bytes by output (%) Example : '`64`' = 100%
`X`	DMX	2 bytes by channel Example : '`C0`' = 192
`T`	Temperature Heating setpoint	Example : '`20.5 22.0 AUTO 18.0`' 1st T° = measure (with software offset) 2nd T° = Heating setpoint value Sensor T° Mode 3rd T° = Heating profile value
`U`	Temperature Cooling setpoint	Example : '`20.5 22.0 HEATING 18.0`' 1st T° = measure (with software offset) 2nd T° = cooling setpoint value Sensor Regulation Mode 3rd T° = cooling profile value
`C`	Infrared Command	Example : Key 1 = '`01`'
`S`	Sound	'`1-32-TUNE-63-03E8`' = Output 1 – 50% - Source Tuner – 99,1000 Mhz (Since card version 5)
`B`	Button	2 bytes(button number) + 2 bytes (`00`=released `01`=pressed)
`P`	Temp. Plage	Example : `12:32:00 21.6` 1st = hh:mm:ss 2nd = setpoint value
`K`	Clocks	Example : `00:38:00 7F 00/01/04 Clock` 1st = hh:mm:ss 2nd = Day mask (b0=sunday, b1=monday, ... b7= disable clock (=1)) 3rd = Name 4th = Type of clock : blank (normal), SUNSET, SUNRISE, RESET

4.4.c) Sample of received strings from your Domintell installation

All strings

Text	Means
PONG	answer from DRS23202/DETH02 after a string "PING"
MOD_VERSION=SER_V0A	answer from DRS23202 after a string "MOD_VERSION" (hexa)
MOD_VERSION=ETH_V01_STK_V01	answer from DETH02 after a string "MOD_VERSION" (hexa)
TE1 6CT25.2 21.0 AUTO 19.5	Heating T° infos of DTEM01 with serial number 0x6C
TE1 6CU25.2 21.0 HEATING 19.5	Cooling T° infos of DTEM01 with serial number 0x6C
TE2 58T20.9 21.0 COMFORT 21.0	Heating T° infos of DTEM02 with serial number 0x58
TE2 58U20.9 28.0 MIXED 28.0	Cooling T° infos of DTEM02 with serial number 0x58
BU1 11O00	Outputs OFF on module DPBU01 with serial number 0x000011
BU2 52O01	led 1 ON on module DPBU02 with serial number 0x000052
BU4 4FO00	Outputs OFF on module DPBU04 with serial number 0x00004F
BU6 8AO00	Outputs OFF on module DPBU06 with serial number 0x00008A
BIR 3A6O00	Outputs OFF on module DBIR01 with serial number 0x0003A6
TRV 73O00	Outputs OFF on module DTRV01 with serial number 0x000073
TRP 151O00	Outputs OFF on module DTRP01 with serial number 0x000151
DIM 19FD 064 0 0 0 0 0 0	Dim 2 = 100% on module DDIM01 with serial number 0x00019F
LED C2O00	Outputs OFF on module DLED01 with serial number 0x0000C2
IS4 7I00	Inputs OFF on module DISM04 with serial number 0x000007
IS8 4F8I10	Key 5 ON on module DISM08 with serial number 0x0004F8
BU1 11I00	Buttons released on module DPBU01 with serial number 0x000011
BU2 52I00	Buttons released on module DPBU02 with serial number 0x000052

BU4 4FI00	Buttons released on module DPBU04 with serial number 0x00004F
BU6 8AI10	Button 5 pressed on module DPBU06 with serial number 0x00008A
BR2 10I00	Buttons released on module DPBR02 with serial number 0x000010
BR4 4FI02	Button 2 pressed on module DPBR04 with serial number 0x00004F
BR6 30O10	Led Output 5 ON on module DPBR06 with serial number 0x000030
B81 11I01	Button 1 pressed on module DPBL01 with serial number 0x000011
B82 52I00	Buttons released on module DPBL02 with serial number 0x000052
B84 4FI00	Buttons released on module DPBL04 with serial number 0x00004F
B86 8AI00	Buttons released on module DPBL06 with serial number 0x00008A
VI1 1I01	Button pressed on DVIP01 with serial number 0x000001
VI2 3I02	Button 2 pressed on DVIP01 with serial number 0x000003
LCD 25I00	Inputs OFF on module DLCD01 with serial number 0x000025
VAR 1O01	Variable 1 True
VAR000001O00	Variable 1 False
VAR 1D64	Variable 1 100%
SYS 2O01	System Variable 2 has value 1
TPV 3O01	shutter 1 : UP on module DTRP02 with serial number 0x000003
D10 1D32	50% on module DOUT10V02 with serial number 0x000001
V24 1O01	shutter 1 : UP on module DTRVBT01 with serial number 0x000001
PBL CO00	Outputs OFF on module DPBTLCD0x with serial number 0x00000C
PBL CT24.0 18.0 AUTO 12.0	Temperature on module DPBTLCD02 with serial number 0x00000C
PBL CB0101	Push Button 1 on DPBTLCD with serial number 0x00000C
PBL CB0100	Release Button 1 on DPBTLCD with serial number 0x00000C

PBL CO00	DPBLCD0xwith serial number 0x00000C outputs are OFF
PBL CO02	2nd DPBLCD0xwith serial number 0x00000C output is ON
PRL CO00	Outputs OFF on module DPBRLCD0x with serial number 0x00000C
PRL CT24.0 18.0 AUTO 12.0	Temperature on module DPBRLCD02 with serial number 0x00000C
PRL CB0101	Push Button 1 on DPBRLCD02 with serial number 0x00000C
PRL CB0100	Release Button 1 on DPBRLCD02 with serial number 0x00000C
PRL CO00	DPBRLCD02 with serial number 0x00000C outputs are OFF
PRL CO02	2nd DPBRLCD02 with serial number 0x00000C output is ON
AMP 3S1-1D-TUNE-6A-0FA0	Output 1, 29%, Tuner, 106.4000MHz on DAMPLI01 with serial 0x03
AMP 3S3-32-AUX1-64-0000	Output 3, 50%, Aux 1, 100.0000MHz on DAMPLI01 with serial 0x03
FAN000001O20	DFAN01 module with serial number 0x000001 is OFF, manual mode
FAN000001O11	DFAN01 with serial number 0x01 is cooling @ speed 1, auto mode
FAN00000100C	DFAN01 with serial number 0x01 is heating @ speed 3, auto mode
FAN000001O32	DFAN01 with serial number 0x01 is cooling @ speed 2, manual mode
DMV000001O01	DMV01 with serial number 0x01 has speed 1 enabled
DMV00000100A	DMV01 with serial number 0x01 has speed 2 and auxiliary 1 output enabled
DMV00000101A	DMV01 with serial number 0x01 has speed 2 and auxiliary 1 and 2 output enabled
DAL 10-08D64	DINTDALI01 #0x10 output 0x08 status @ 100%
I10000005D32	Input = 50% on DIN10V02 with serial number 0x000005
DMX 1F-2-00EB000000000000	String with 2nd '-' is obsolete since v11(DETH02) & v16(DRS23202) Device 2 connected to DDMX01 module with serial number 0x00001F has its 2nd channel set to 234
DMX 1F-2X00EB000000000000	
MBD 201T22.7 23.0 AUTO 23.0	Heating T° infos of ModBus Device with serial

	number 0x201
`MBD 201U22.7 26.0 OFF 26.0`	Cooling T° infos of ModBus Device with serial number 0x201
`MBD 201D 3 2`	Device specific values for ModBus Device with serial number 0x201
`CLK 2K08:05:00-7F-00/00/00-Clock[SUNRISE]`	Clock 2 is an astronomical sunrise clock set (this week) to 8h05m00s all weekdays
`TPR 2Range N°2`	Profile 2 is named 'Range N°2'
`TPL 8P15.5-02:45:00`	Setpoint of Range 8 will be 15.5°C from 2h45m00s
`STA 1STU BRU[FM=64-1770]`	Station 1 « STU BRU » @ FM 100,6000MHz
`!! PLEASE UPGRADE DETH02 FIRMWARE`	This string means that DETH02 has an incompatible version regarding the current OS version in the Master/DGQG01. This can also occur if status of a new module's type is received by DETH02/DRS23202 and is not handled by its firmware. Bad/missing information can be sent by DETH02 until its firmware is updated.

4.4.d) Decoding APPINFO

- Warnings/Errors
  They starts with an exclamation mark ("`!`") and must be shown to user and ask him to contact Domintell support.
  - `!! PLEASE UPGRADE DRS23202 FIRMWARE ≥ 18 !!` or `!! PLEASE UPGRADE DETH02 FIRMWARE >= 17 !!`
    DRS23202/DETH02 needs an update of its firmware to be able to decode information send by master/DGQG01 or new references of modules. Customer's application (i.e. Smartphone App) can still work but some status/commands can not anymore until the firmware is updated.
  - `! PLEASE RESTART MASTER 0x???????? !`
    Where `????????` is the serial number of the module that is not in the module table of the DRS23202/DETH02 module. The DRS23202/DETH02 module was not connected to the bus when (1) the application has been sent to master/DGQG01, (2) the master has been restarted and the DRS23202/DETH02 module did not receive the new table. Or The specified module has been added after the bus has been scaned by the master/DGQG01.

- APPINFO line gives information about the DAP/configuration file :
  "`APPINFO (PROG M 33.0 00/00/00 00h00 Rev=3 CP=UTF-8) => Smith_v33_v7.dap :`"
  - `PROG M 33.0 … Rev=0` : stands for DAP file version 33.0.0.
  - Date/Hour field is deprecated from version 31.
  - `CP=UTF-8` (only starting from OS version 1.27.06) : specifies the Windows charset (CP=1252) Unicode charset used for non-ASCII

accentuated characters. CP1252 is also known as Windows-1252. See Windows code page list on Wikipedia for more information.
  - `Smith_v33_v7.dap` is the application name (truncated to 32 characters).

- Room/floor information where the input/output/memo/ambiance are located are given just after the name of the item. and will look like `[House|_floor_/_room_]` where `_floor_` and `_room_` are replaced by the real floor and room names (without the underscores "`_`")

- Some devices has extra type information like DINTDALI (`DAL`) IO's or DDMX01 (`DMX`) IO's just after the room/floor information (`[House|Ground|Living]`)
  - DINTDLAI01 : `[TYPE=xx]` specifies the type of the DALI slave where "`xx`" can be :
    - `TL` : Device Type 0 for fluorescent lamps (IEC 62386-201)
    - `ER` : Device Type 1 for self-contained emergency lighting (IEC 62386-202)
    - `DISC` : Device Type 2 for discharge lamps (IEC 62386-203)
    - `LOWV` : Device Type 3 for low voltage halogen lamps (IEC 62386-204)
    - `INCA` : Device Type 4 for supply Voltage controller for incandescent lamps (IEC 62386-205)
    - `DC` : Device Type 5 for conversion from digital into D.C. voltage (IEC 62386-206)
    - `LED` : Device Type 6 for LED modules (IEC 62386-207)
    - `SW` : Device Type 7 for switching function (IEC 62386-208)
    - `RGB` : Device Type 8 for colour control (IEC 62386-209)
      Please note that all device types can not be handled by DINTDALI01 firmware.

  - DDMX01 output : `[x CHANNELS]` specifies how much channels are configured for this DDMX01 output where "`x`" can be a value between 1 to 8

- The application (APPINFO) is fully retrieved when the string beginning with "`END APPINFO`" is received.

- Example of received strings with APPINFO command :

```
!! PLEASE UPGRADE DRS23202 FIRMWARE >= 24 !!
!! PLEASE UPGRADE DETH02 FIRMWARE >= 25 !!
APPINFO (PROG M 1.27 04/11/16 09h28 Rev=3) => TEST_APPINFO.dap :
FRO     1 : 1
RS2     2[VERS=0x10]Interface protocole RS[House||]
ET2    B6[VERS=0x0B]MOD DETH02[House||]
BIR   4C9-1BIR 1[House|1st floor|living]
BIR   4C9-2BIR 2[House|1st floor|living]
BIR   4C9-3BIR 3[House|1st floor|kitchen]
BIR   4C9-4BIR 4[House|1st floor|kitchen]
BIR   4C9-5BIR 5[House|2nd floor|]
BIR   4C9-6BIR 6[House||]
BIR   4C9-7BIR 7[House||]
BIR   4C9-8BIR 8[House||]
TRV   3E9-1TRV 1[House||]
TRV   3E9-3TRV 2[House||]
TRV   3E9-5TRV 3[House||]
TRV   3E9-7TRV 4[House||]
```

```
DMV     1-1Output DMV01[House||]
DMV     1-1Output DMV01 2[House||]
DMV     1-1Output DMV01 3[House||]
DMV     1-1Auxiliary 1[House||]
DMV     1-1Auxiliary 2[House||]
PBL   E6C-1Input PB 1[House||][NOLINK]
PBL   E6C-2Input PB 2[House||][NOLINK]
PBL   E6C-3Input PB 3[House||][NOLINK]
PBL   E6C-4Input PB 4[House||][NOLINK]
PBL   E6C-7T° sensor DPBTLCD0x[House||]
PBL   E6C-8Led PB 1[House||]
PBL   E6C-9Led PB 2[House||]
PBL   E6C-ALed PB 3[House||]
PBL   E6C-BLed PB 4[House||]
LT4     1-5T° sensor DTSC04[House||]
LT4     1-6IR sensor DTSC04[House||]
LT4     1-BOutput DTSC04 1[House||]
LT4     1-COutput DTSC04 2[House||]
LT4     1-DOutput DTSC04 3[House||]
LT4     1-EOutput DTSC04 4[House||]
LT4     1-15Lock[House||]
BU6   24B-1Input B6 1[House||][PUSH=LONG]
BU6   24B-2Input B6 2[House||][PUSH=LONG]
BU6   24B-3Input B6 3[House||][NOLINK]
BU6   24B-3Input B6 3[House||][NOLINK]
BU6   24B-4Input B6 4[House||][PUSH=SHORT]
BU6   24B-5Input B6 5[House||][PUSH=SHORT]
BU6   24B-6Input B6 6[House||][NOLINK]
BU6   24B-7LED B6 1[House||]
BU6   24B-8LED B6 2[House||]
BU6   24B-9LED B6 3[House||]
BU6   24B-ALED B6 4[House||]
BU6   24B-BLED B6 5[House||]
BU6   24B-CLED B6 6[House||]
DIM   21B-1DIM 1[House||]
DIM   21B-2DIM 2[House||]
DIM   21B-3DIM 3[House||]
DIM   21B-4DIM 4[House||]
DIM   21B-5DIM 5[House||]
DIM   21B-6DIM 6[House||]
DIM   21B-7DIM 7[House||]
DIM   21B-8DIM 8[House||]
TSB   236-5T° sensor Touch[House||]
TSB   236-6IR sensor Touch[House||]
TRP   691-1TRP 1[House||]
TRP   691-2TRP 2[House||]
TRP   691-3TRP 3[House||]
TRP   691-4TRP 4[House||]
BU2     9-1Input B2 1[House||][PUSH=SHORT]
BU2     9-2Input B2 2[House||][NOLINK]
BU2     9-3LED B2 1[House||]
BU2     9-4LED B2 2[House||]
TE1   9DE-1T° sensor T1[House||]
V24     A-1TRV BT[House||]
I10     5-1Input 0-10V [House||]
AMP   105-1HP 1[House||]
AMP   105-2HP 2[House||]
AMP   105-3HP 3[House||]
AMP   105-4HP 4[House||]
FAN   267-1DFAN[House||]
FAN   268-1DFAN[House||]
DMR     3-1DMR 1[House||]
DMR     3-2DMR 2[House||]
DMR     3-3DMR 3[House||]
```

```
DMR     3-4DMR 4[House||]
DMR     3-5DMR 5[House||]
DMX    91-1DMX Output 1 RGBI[House||][4 CHANNELS]
DMX    91-1-CH1:Chan. R[R 0x00-0xFF]
DMX    91-1-CH2:Label G[G 0x00-0xFF]
DMX    91-1-CH3:Chan. B[B 0x00-0xFF]
DMX    91-1-CH4:Chan. I[I 0x00-0x64]
DMX    91-2DMX Output 2 II[House||][2 CHANNELS]
DMX    91-2-CH1:Chan. 1[I 0x00-0xFF]
DMX    91-2-CH2:Chan. 2[I 0x00-0xFF]
DMX    91-3DMX Output 3 I[House||][1 CHANNELS]
DMX    91-3-CH1:Chan. 1[I 0x00-0xFF]
DAL    10-01TL #12345678-1[House||][TYPE=TL] (!DALI Out number = 2
digits!)
DAL    10-02LED #87654321-2[House||][TYPE=LED]
DAL    10-03PHASE #87654321-2[House||][TYPE=INCA]
B81     2-1Button 1[House|Floor|Room]
B82    12-2Button 2[House||]
B84    36-3Button 3[House||]
B86    72-4Button 4[House||]
B86    89-7LED B6 1[House||]
B84   347-6LED B4 2[House||]
B82    84-3LED B2 1[House||]
B81    39-2LED B1 1[House||]
PRL   E6C-1PBRLCD Input 1[House||][NOLINK]
PRL   E6C-2PBRLCD Input 2[House||][NOLINK]
PRL   E6C-3PBRLCD Input 3[House||][NOLINK]
PRL   E6C-4PBRLCD Input 4[House||][NOLINK]
PRL   E6C-7PBRLCD T° sensor[House||]
PRL   E6C-8PBRLCD Led 1[House||]
PRL   E6C-9PBRLCD Led 2[House||]
PRL   E6C-APBRLCD Led 3[House||]
PRL   E6C-BPBRLCD Led 4[House||]
VAR     1My variable[House|Floor|Room][BOOL]
VAR     2My variable 2[House|Floor|Room][VALU,00->100,LOOP]
SYS     0Presence simulation[House||][BOOL]
SYS     1T° mode[House||][VALU,1-2-5-6,LOOP]
SYS     2Regulation mode[House||][VALU,00->03,LOOP]
SYS     9Day[House||][BOOL][READONLY]
MEM     1Memo 1[House||][MIX][REF=BIR   4C9-1]
MEM     2Memo 2[House||][SHUTTERS][REF=TRV   3E9-1]
MEM     3Memo 3[House||][DIMMERS][REF=DIM   21B-1]
MEM     4Memo 4[House||][SOUND][REF=AMP   105-1]
MEM     5Memo 5[House||][FAN][REF=FAN   267-1]
SFE     1Sfeer 1-Scene 1[House||]
SFE     2Sfeer 1-Scene 2[House||]
ZON     1Zone 1[House||]
CLK     1K00:38:00-7F-04/01/00-Clock
CLK     2K08:05:00-7F-00/00/00-Clock[SUNRISE]
CLK     3K00:00:00-7F-00/00/00-Clock[RESET]
CLK     4K18:02:00-7F-00/00/00-Clock[SUNSET]
TPR     1Range N°1
TPL     0P12.0-00:00:00
TPL     1P26.5-05:00:00
TPL     2P12.0-07:00:00
TPL     3P 5.0-13:45:00
TPL     4P12.0-15:45:00
TPL     5P20.0-20:15:00
TPL     6P12.0-22:15:00
TPR     2Range N°2
TPL     7P12.0-00:00:00
TPL     8P15.5-02:45:00
TPL     9P12.0-04:45:00
TPL     AP26.0-08:30:00
```

```
TPL     BP12.0-10:30:00
TPL     CP30.0-16:30:00
TPL     DP12.0-18:30:00
STA     1STU BRU[FM=64-1770]
STA     2PURE FM[FM=60-1770]
CAM     1Axis Cam01[AXIS][IP=192.168.0.2]
CAM     2Cam DVIP01[DVIP][DHCP][IP=192.168.0.3][JPG=http://192.168.0.3:80/
jpg/image.jpg]
CAM     3Cam
DVIP02[DVIP][IP=192.168.0.4][JPG=http://192.168.0.4:80/jpg/image.jpg]
END APPINFO - Send "HELP" from ETH.
Datasheet @ www.domintell.com => Pro - support@domintell.com
```

4.5. New generation LightProtocol

4.5.a) Overview

These commands/strings can be sent to Domintell2 system and are executed without doing any links (Automatic LightProtocol).
All newly released modules will follow the "*new generation input LightProtocol*". This new version is needed to handle these new complex modules that have several types of input and/or output (e.g. DGQG02 has TOR outputs, TRV output, 0-10V outpus, ...) that can not be handled with the legacy LightProtocol.
Frame generator of LightProtocol server for new generation modules has been completely reviewed to be parsed more easily.

4.5.b) Modules using new generation input LightProtocol

- DGQG02
- DGQG03
- DENV01
- DENV02
- DMOV06
- DWIND01
- DDMX02

4.5.c) General new generation frame format

All numbers are now expressed, by default, **in decimal** (even the serial number). If you want to use hexadecimal notation, you need to prefix value with "0x". So set an ouput 50% you can use `50` or `0x32`.

```
<Module type>/<serial number>/<data frame>
```

- `<Module type>` : Module identifier. See 4.1 Abreviation of Modules' type for more informations.
- *`<serial number>`* : is expressed in declmal.
- `<data frame>` can contain following special characters:
  - `/` (slash) : field separator.
  - `#` (number sign) : input/output separator.
  - `|` (pipe) : parameter separator.

4.5.d) APPINFO frame format (*LightProtocol server* to *LightProtocol client*)

This chapter only covers lines generated for new generation modules. Please first read the chapter for the legacy APPINFO (4.4.d Decoding APPINFO).
See 4.5.c General new generation frame format for the start of the frame.

```
<Module type>/<serial number>/<IO type>/<IO offset>/<IO name>/<module version>/<house location>/<extra IO informations>
```

- `<Module type>` : Module identifier. See 4.1 Abreviation of Modules' type for more informations.
- *`<serial number>`* : is expressed in declmal.
- *`<IO type>`* : can be one of the following given in seuction 4.5.j IO type list and data format.
- *`<IO offset>`* : 1-based offset for the given IO type.
- *`<IO name>`* : name of the IO.

- `<module version>` : Version of the module in dotted decimal string.
- `<house location>` : location in the house. Room/floor are seperated by pipes.
- `<extra IO informations>` : For *TypeInputIo(2)* IO type, it will be type of push configured or if there are links associated to this input. See 4.5.k Extra informations in APPINFO.

4.5.e) Example of APPINFO frame

- `QG2/12(/A)/2/1/Hall lights/1.8.0/Ground floor|Hall/0`
  - DGQG02
  - Decimal serial number 12
  - Input IO/Push button (2=TypeInputIo)
  - Input number 1
  - with the name "Hall lights"
  - version of DGQG02 is 1.8.0
  - IO is located in Ground Floor and Hall room
  - No link programmed (0)
- `QG2/12(/A)/2/2/Living lights/1.8.0/Ground floor|Living/2`
  - DGQG02
  - serial number 12
  - (APPINFO Frame)
  - Input IO/Push button (2=TypeInputIo)
  - Input number 2
  - with the name "Living lights"
  - version of DGQG02 is 1.8.0
  - IO is located in Ground Floor and Living room
  - Long and short pushes are enabled (links programmed) (2)

4.5.f) Status frame format (*LightProtocol server* to *LightProtocol client*)

See 4.5.c General new generation frame format for the start of the frame.

```
<Module type>/<serial number>/<IO type>/<IO offset>/<data1>#<data2>#...
```

- `<Module type>` : Module identifier. See 4.1 Abreviation of Modules' type for more informations.
- `<serial number>` : is expressed in declmal.
- `<IO type>` : can be one of the following given in section 4.5.j IO type list and data format.
- `<IO offset>` : 1-based offset for the given IO type.
- `<datax>` : Status starting from IO with offset `<IO offset>`. If several statuses are provided (separated using the number sign (`#`)), IO offset should be increases each the time number sign (`#`) is decoded and associate the value following this number sign (`#`) to this "increased" IO offset. The format of the data depends on `<IO type>`, see section 4.5.j IO type list and data format for more informations.

4.5.g) Examples of status frame

- `QG2/12/2/1/2#2#1#2#2#2#2#2#2#2#2#2`
  DGQG02, serial 12, TypeInputIo, from input 1 (IN1) to input 12 (IN12), all inputs are released except IN3 which is pressed.

This kind if frame is received in reply to the PING command (5.7 Refresh statuses).

- `QG2/12/2/11/1`
  Status of inputs (2 = TypeInputIo) and starts from input 11.
  Input 11 has just changed and is pressed.
- `QG2/12/1/1/0`
  DGQG02, serial 12, TypeTorIo, output 1 (OUT1), off state
- `QG2/12/1/8/1`
  DGQG02, serial 12, TypeTorIo, output 8 (OUT8), on state
- `QG2/1/23/1/45#0`
  DGQG02, serial 12, TypeOut10VIo, from 0-10V output 1, « +1 » output = 0 % (0V) and « +2 » output = 45 % (4,5V)
- `QG2/1/23/2/0`
  DGQG02, serial 12, TypeOut10VIo, 0-10V output 2 (+2), 0 % (0V)
- `QG2/1/23/1/45`
  DGQG02, serial 12, TypeOut10VIo, 0-10V output 1 (+1), 45 % (=4,5V)
- `MV6/3/34/1/1`
  DMOV06, serial 3, TypeMovIo, input 1, start detection
- `MV6/3/34/1/2`
  DMOV06, serial 3, TypeMovIo, input 1, end of detection
- `EV1/3/37/1/56.6`
  DENV01, serial 3, TypeHumidityIo, input 1, 56.6% RH
- `EV1/3/38/1/996.4`
  DENV01, serial 3, TypePressureIo, input 1, 996.4 hPa
- `EV1/3/36/1/1798`
  DENV01, serial 3, TypeLuxIo, input 1, 1798 lux
- `EV2/7/39/1/550.6`
  DENV02, serial 7, TypePressureIo, input 1, 550.6 hPa
- `EV2/7/37/1/54.2`
  DENV02, serial 7, TypeHumidityIo, input 1, 54.2% RH
- `EV2/7/8/1/22.1|24.0|AUTO|21.0|25.0|HEATING|27.0`
  DENV02, serial 7, TypeSensorIo, input 1,
  current temperature is 22.1°C,
  active heating setpoint is 24.0°C,
  current temperature mode is AUTO,
  profile heating setpoint is 21.0°C,
  active cooling setpoint is 25.0°C,
  current regulation mode is HEATING,
  profile cooling setpoint is 27.0°C

4.5.h) Command frame format (*LightProtocol client* to *LightProtocol server*)

See 4.5.c General new generation frame format for the start of the frame.

```
<Module type>/<serial number>/<IO type>/<IO offset>/<cmd1>[|<data1.1>|
<data1.2>|...]#<cmd2>[|<data2.1>|...]#...
```

- `<Module type>` : Module identifier. See 4.1 Abreviation of Modules' type for more informations.
- `<serial number>` : is expressed in declmal.
- `<IO type>` : can be one of the following given in section 4.5.j IO type list and data format.
- `<IO offset>` : 1-based offset for the given IO type.
- `<cmdx>` : Command to perform starting from IO with offset `<IO offset>`. If several commands are provided (separated using the

number sign (#)). The format of the data depends on <IO type> and <cmd>, see section 4.5.j IO type list and data format for more informations.

- <datax.y> : argument/extra data associated to the <cmdx>. Data are seperated from each others and its related command by a pipe (|). This field is optional.

4.5.i) Example of command frame

- Toggle the 8th relay (TypeTorIo = 1) of DGQG02 s/n 95 : QG2/95/1/8/1
- Turn on the 1st relay (TypeTorIo = 1) of DGQG02 s/n 95 : QG2/95/1/1/2
- Turn off the 3rd relay (TypeTorIo = 1) of DGQG02 s/n 95 : QG2/95/1/3/3
- Set 1st 0-10V output (TypeOut10VIo = 23) of DGQG02 s/n 95 to 90% : QG2/95/23/2/5|90
- Move TRV output (TypeTrvIo = 6) of DGQG02 s/n95 up : QG2/95/6/1/10
- Set 3rd channel (TypeDmxIo = 25) of slave 5 of of DDMX02 s/n 15 to 250 : DX2/15/25/5/2/71#0x04|0|0|250 (All channels below the highest bit set must be declared in value list (in this case set to 0 but will be ignored by DGQG)

4.5.j) IO type list and data format

Decimal number used to define the category of the IO :

- TypeTorIo (relay) = 1
  - Unit : none
  - Possible status : 0-1
  - Possible command : 1-3
    - 1 = toggle
    - 2 = on
    - 3 = off
- TypeInputIo (push-button input) = 2
  - Unit : none
  - Possible status : 0-4
    - 0 = Unknown state
    - 1 = Start of short push (currently pressed)
    - 2 = End of short push (currently released)
    - 3 = Start of long push (currently pressed)
    - 4 = End of long push (currently released)
  - Possible command : 1-4
    - 1 = Execute links associated to start of short push
    - 2 = Execute links associated to end of short push
    - 3 = Execute links associated to start of long push
    - 4 = Execute links associated to end of long push
- TypeDimmerIo (dimmer output) = 3
  - Unit : percent (%)
  - Possible status : 0-100
  - Possible command : 1, 2, 3, 5
    - 1 = toggle
    - 2 = on

        - 3 = off
        - 5 = set to value given in `<data>`.
- TypeTrvIo (shutter output) = 6
    - Unit : none
    - Possible status : 0-5
        - 0 = unknown state = 0
        - 1 = stopped (last moving side unknown)
        - 2 = moving up
        - 3 = moving down
        - 4 = stopped and last move was up
        - 5 = stopped and last move was down
    - Possible command : 3, 10, 11
        - 3 = stop shutter
        - 10 = move shutter up
        - 11 = move shutter down
- TypeSensorIo (Temperature sensor) = 8
    - Unit : celcius degree (°C)
    - Possible status : contains several data seperated by pipe '|' ! <Meas. Temp> | <Active Heat. SP> | <Temp. Mode>| <Heat Prof. SP> | <Active Cool. SP> | <Regul. Mode> | <Cool Prof. SP>
      For example: 22.1|24.0|AUTO|21.0|25.0|HEATING|27.0
    - Possible command : 1, 2, 3, 5
        - 1 = toggle
        - 2 = on
        - 3 = off
        -
- TypeOut10VIo (0-10V output) = 23
    - Unit : percent (%)
    - Possible status : 0-100
    - Possible command : 1, 2, 3, 5
        - 1 = toggle
        - 2 = on
        - 3 = off
        - 5 = set to value given in `<data>`.
- TypeDmxIo (DMX output) = 25
    - Unit : none
    - Channel seperator : `|`
    - Possible status : 0-255
    - Possible command : 1, 2, 3, 71
        - 1 = toggle
        - 2 = on
        - 3 = off
        - 71 = set to value accroding informations given in `<data>` with `<data>` = `<mask> | <value1> | <value2> | ...`
            - mask : `value1` is linked to bit 0 of `mask`, `value2` is linked to bit 1 of `mask`, ...
              If highest bit set in mask is the bit 5, you need to provide 6 values even if some bits between bit 5 and bit 0 are null. So if you want to set (virtual) RGB intensity channel and

white channel of a RGBW slave to 200 and 100 respectively, data will look like `0x18|0|0|0|100|200`.
    - 0x07: set all RGB channels of RGB, RGBI and RGBW slaves. 3 values are needed. `value1` = Red channel, `value3` = Blue channel. Example : `0x07|100|255|100`.
    - 0x08: set Intensity channel of RGB (virtual) and RGBI. 4 values are needed. `value4` will be used. `value1` to `value3` will be ignored. Example `0x08|0|0|0|255`.
    - 0x10: set (virtual) RGB intensity channel of a RGBW slave. Only value5 will be used. Example `0x10|0|0|0|0|255`.
- TypeDali (DALI output) = 29
- TypeMovIo (motion detector input) = 34
    - Unit : none
    - Possible status : 0-2
        - 0 = Unknown state
        - 1 = Start of detection (or move in progress)
        - 2 = End of detection (no move detected anymore)
    - Possible command : 1-2
        - 1 = Execute links associated to start of detection
        - 2 = Execute links associated to end of detection
- TypeLuxIo (light sensor input) = 36
    - Unit : lux (lux)
    - Possible status : 0-16.000 (can be higher)
    - Possible command : None
- TypeHumidityIo (humidty sensor input) = 37
    - Unit : percent (% RH)
    - Possible status : 0-100
    - Possible command : None
- TypePressureIo (atmospheric pressure sensor input) = 38
    - Unit : hectopascal (hPa)
    - Possible status : 300-1100
    - Possible command : None
- TypeCo2Io (Carbon dioxide sensor input) = 39
    - Unit : parts-per-million (ppm)
    - Possible status : 0-40000
    - Possible command : None

4.5.k) Extra informations in APPINFO

- TypeInputIo (2) : `/<type_of_link>`
    - `<type_of_link>`:
        - 0 = No link
        - 1 = Short push only
        - 2 = Short and long push
- TypeDmxIo (25) : `/<number_of_channel>|<dmx_type>`
    - `<number_of_channel>` : number of channels defined for this DMX slave/output
    - `<dmx_type>` : defines the mapping of channels
        - 0 = misconfigured DMX slave/output. Please check configuration in GoldenGate !

- 1 = RGB : first channel = red, second channel = green, third channel = blue.
- 2 = RGBI : first channel = red, second channel = green, third channel = blue, fourth channel = intensity.
- 3 = RGBW : first channel = red, second channel = green, third channel = blue, fourth channel = white
- 4 = single channel (of any color)
- 5 = multiple channels (of any color)

4.6. Custom input ASCII strings (sent to Domintell system).

Custom texts or frame from external devices/services are not handled.

4.7. Custom output ASCII strings (sent to your device).

Custom texts or frame to external devices/services are not handled.

5. How to use LightProtocol with your own application

5.1. Related modules

This chapter convers New Generation Ethernet modules like DNET01, DGQG02, DGQG03, DGQG04 with control mode (ETH02 emulation) enabled. For old DETH02 module, please read DS_RS232_ETH_Interfaces_v1_27_08.pdf.

5.2. Tools

New generation Ethernet Domintell modules now use Secured WebSocket (wss : data are encrypted in encapsulated frame over HTTP protocol). For now, there is only one out-of-box tool available under Linux : wscat.

5.2.a) wscat

```
$ wscat -n -c wss://<ip>:17481
> connected (press CTRL+C to quit)
< INFO:Waiting for LOGINPSW:INFO
> LOGINPSW@:
< INFO:Session opened:INFO
> APPINFO
<   APPINFO   (PROG   M   30.9   06/02/17   09h19   Rev=13   CP=UTF8)   =>
HOUSE_v300912_v2 :
NT1     1[VERS=0x04]Module DNET01[House||]
IS8   5B1-1BP sirène[House||][PUSH=SHORT]
IS8   5B1-2ISM8 2[House||][NOLINK]
IS8   5B1-3ISM8 3[House||][NOLINK]
IS8   5B1-4ISM8 4[House||][NOLINK]
IS8   5B1-5ISM8 5[House||][NOLINK]
IS8   5B1-6ISM8 6[House||][NOLINK]
IS8   5B1-7ISM8 7[House||][NOLINK]
IS8   5B1-8ISM8 8[House||][NOLINK]
END APPINFO
Datasheet @ www.domintell.com => Pro - support@domintell.com
> LOGOUT
< INFO:Closing session:INFO
disconnected
```

5.3. Password handling

Accounts must be first created using GoldenGate !

5.3.a) Password algorithm

The mechanism uses standard encryption algorithm. It uses salted password hashed. This hashed password is itself hashed with nonce generated by the Domintell module at socket creation. SHA-512 is used to hash data.

sha512(sha512(password + salt) + nonce)

5.3.b) Glossary

- *hash*: It is the result of a mathematical algorithm that maps data of arbitrary size to a bit string of a fixed size (a hash) and is designed to be a one-way function, that is, a function which is infeasible to invert.
- *SHA-512*: It is a mathematical algorithm that maps data of arbitrary size to a bit string of a fixed size of 64 bytes (a hash) and is designed to be a one-way function, that is, a function which is infeasible to invert.

- *salt*: random data that is used as an additional input to a one-way function that hashes a password or passphrase.
- *nonce*: an arbitrary number that can be used just once in a cryptographic communication.

5.4. Open a session

Depending of the version of the Ethernet module, when WebSocket are used, Welcome message differs and tells which mechanism should be used:

- `INFO:Waiting for LOGINPSW:INFO`
  This is the first generation of login mechanism. There is no handling of user/password and the following command must be sent to open a session: `LOGINPSW@:`

- `INFO:Waiting for LOGINPSW:NONCE=19228322921537008311:INFO`
  This is the last generation of login mechanism that supports user/password. and the following procedure should be used to connect.

Consider we have the registered user "toto" with password "azerty". Here is the procedure to log in:

1. When a Secured WebSocket is opened, A text message is sent by Domintell module:

```
connected (press CTRL+C to quit)
< INFO:Waiting for LOGINPSW:NONCE=9301906811536867321:INFO
```

2. Request the salt associated to the user

```
> REQUESTSALT@toto
<
INFO:REQUESTSALT:USERNAME=toto:NONCE=9301906811536867321:SALT=1007182019:INFO
```

3. You have now all data needed to compute log in token. Salt the password by concatenating password to salt (<password> + <salt>) :

salted password = "*azerty1007182019*"

4. Hash the salted password :

SHA512("azerty1007182019") =
"df6b9fb15cfdbb7527be5a8a6e39f39e572c8ddb943fbc79a943438e9d3d85ebfc2ccf9e0eccd9346026c0b6876e0e01556fe56f135582c05fbdbb505d46755a"

Pay attention that the hash must be converted into an hexa human readable string before appending the nonce alphanumeric string.

5. Prepend the nonce to the hashed salted password (<hashed salted password> + <nonce>) :

"df6b9fb15cfdbb7527be5a8a6e39f39e572c8ddb943fbc79a943438e9d3d85ebfc2ccf9e0eccd9346026c0b6876e0e01556fe56f135582c05fbdbb505d46755a" + "9301906811536867321"=
"df6b9fb15cfdbb7527be5a8a6e39f39e572c8ddb943fbc79a943438e9d3d85ebfc2ccf9e0eccd9346026c0b6876e0e01556fe56f135582c05fbdbb505d46755a9301906811536867321"

5. Hash the nonce and the hashed salted password:
SHA512("930190681153686732 1df6b9fb15cfdbb7527be5a8a6e39f39e572c8ddb943fbc79a943438e9d3d85ebfc2ccf9e0eccd9346026c0b6876e0e01556fe56f135582c05fbdbb505d46755a") = "c83b274c82c98965b558762e9f05e5556de17712c9c74ca372eb64523963ec729def36c384e7ef8a19a8c7373ac3cc4f10d567bc3b2110fec80c199fcea08b01"

6. Build LOGINPSW command:

```
>
LOGINPSW@toto:c83b274c82c98965b558762e9f05e5556de17712c9c74ca372eb64
523963ec729def36c384e7ef8a19a8c7373ac3cc4f10d567bc3b2110fec80c199fce
a08b01
< INFO:Session opened:INFO
> LOGOUT
< INFO:Closing session:INFO
disconnected
```

5.4.a) Error messages

- No user/account in database

```
connected (press CTRL+C to quit)
< ERROR:User database empty. Connect first with GoldenGate:ERROR
< INFO:Session timeout:INFO
disconnected
```

This is the default behaviour when the module is connected for the first time on a Domintell installation. No connection is allowed until a user account is created using the configuration software GoldenGate.

- Wrong credentials

This error occurs when username, password or token is invalid.

```
connected (press CTRL+C to quit)
< INFO:Waiting for LOGINPSW:NONCE=8425756741536921788:INFO
>
LOGINPSW@toto:c83b274c82c98965b558762e9f05e5556de17712c9c74ca372eb64
523963ec729def36c384e7ef8a19a8c7373ac3cc4f10d567bc3b2110fec80c199fce
a08b01
< ERROR:Invalid credentials:ERROR
disconnected
```

In this case, the credentials are rejected because the same token generated for account creation is reused while nonce has changed ! Invalid credentials

- Session not opened

No commands are accepted until a session is opened.

```
connected (press CTRL+C to quit)
< INFO:Waiting for LOGINPSW:NONCE=10671983681536923762:INFO
> PING
< ERROR:Invalid command. Use REQUESTSALT@<username> and
LOGINPSW@<username>:<hashedpsw>:ERROR
disconnected
```

PING has been sent while no session is opened.

5.5. Download list of modules (TODO !)

```
> APPINFO
< !! PLEASE UPGRADE DRS23202 FIRMWARE >= 18 !!
< APPINFO (PROG M 1.24 16/06/14 09h44 Rev=0) =>
DOMINT_v12400_v02.dap :
< FRO     1 : 1
< ET2     1[VERS=0x14]MOD DETH02[Maison||]
```

```
< …
< END APPINFO - Send "HELP" from ETH.
< Datasheet @ www.domintell.com => Pro - support@domintell.com
```

If red message is also received, you have to inform the customer that the DETH02 module must be updated (by contacting technical support of Domintell) and also inform that some functionnalities may not work correctly.

5.6. Keep session open

To keep session opened, you have to send one command (or LightProtocol string) to LightProtocol server. The best way is to use HELLO command and sent it each 50 seconds. PING command should be avoid to keep a session opened as it will generate a lot of trafic on Domintell Bus and takes ressources in Master (DGQG02/DGQG04/...).

```
> HELLO
< INFO:World:INFO
```

If HELLO command is not received by the DETH02 before its timeout expiration, the following text will be sent:

```
< INFO:Session timeout:INFO
```

A new command has been implemented to disable timeout. Take care that session will be closed only when WebSocket will be closed. In this case, HELLO command is no more needed.

```
> TIMEOUT=0
< INFO:Timeout disabled. Socket will never be closed unless you send
LOGOUT or the connection is lost !:INFO
```

5.7. Refresh statuses

PING command must be used carefully. Generally, use it after a LOGIN (if your application has already been configured using APPINFO).

```
> PING
< PONG
< QG2/12/1/1/0#0#0#0#0#0#0#1
QG2/12/23/1/45#0
QG2/12/2/1/2#2#2#2#2#2#2#2#2#4#2
QG2/12/6/1/5
BIR  101FO00
VAR     1O00
VAR     2D02
SYS     0O00
SYS     9O01
```

There is not string/flags to notify end of list of statuses.

5.8. Close session before exiting the application

If your application is closed or background, it is better to send the LOGOUT command to allow other applications/devices to use DGQG02/DGQG04/DNET01/...

```
> LOGOUT
< INFO:Session closed:INFO
```