

## « RS485/RS422 Transmission » leaflet for « A.M.I » products

- |   |                |
|---|----------------|
| - <b>J3000 RS 485</b> MODBUS / JBUS alarm annunciator panel | Version v1.25E |
| - <b>J2000 RS 485</b> MODBUS / JBUS indicator panel         | Version v2.06  |
| - <b>J2x00 RS 485</b> MODBUS / JBUS indicator panel         | Version v2.06  |
| - <b>MO601</b> MODBUS / JBUS Input-Output module            |                |

**A.M.I**

3, rue de la garenne- ZI Vernon  
F27950 St MARCEL

Tél : 33 (0)2-32-514-716  
Fax : 33 (0)2-32-211-373

E-mail : [ami.com@wanadoo.fr](mailto:ami.com@wanadoo.fr)

[Http://www.ami-control.com](http://www.ami-control.com)

## Summary :

### A) WIRING FEATURES AND COMPONENTS:

#### 1 °) The various bus modules available :

- a) Alarm annunciator panel J3000
- b) Indicator panels J2000RS485 and J2x00RS485
- c) MO601 I/O RS 485 modules

#### 2 °) Diferrent solutions :

- a) Centralization of simple alarms
- b) Sub station linked by bus to automaton or supervision
- c) Sub stations linked by bus to G0700 administrator
- d) Sub stations linked by bus to G0700 with PC / SCREEN

### B) J3000 RS 485

#### 1 °) J3000 functions

- a) Principle :
- b) Panel inputs activation by bus
- c) Additive functions :
  - The memorization of the transmission parameters and RS485 configuration
  - Running bus control
  - The internal meter
  - The "historic" buffer
  - Visualization of the J3000's setting

#### 2 °) Display and modification of the setting parameter on J3000

#### 3 °) Memory adress available on J3000 RS485

- a) Reading of led state
- b) Reading / Writing in the bus input memory
- d) Reading of the channel state
- e) Reading the channel setting parameter
- f) Reading/writing on the KL alarm relay
- g) Writing for the led test
- h) Writing Reset on panel
- i) Reading Reset operator
- j) Standard parameter

#### 4 °) Fonctionning of the "events"buffer

#### 5 °) Software added on J3000 RS 485

### C) J2000 RS 485 AND J2X00 RS 485

#### 1 °) Function of J2000 RS 485 and J2x00 RS 485

- a) Principle
- b) Visualization of the of J2000 parameter

#### 2 °) Memory adress available on J2000 RS485 and J2x00RS485

- a) Reading / Writing of the various states.
- b) Reading / Writing standard setting

#### 3 °) Different versions of the J2000RS485 and J2x00RS485 program

### D) MO 601 RS485 MODULE

#### 1 °) Functions of the I/O module

#### 2 °) Dimensions and Connections

#### 3 °) Channel wiring

#### 4 °) "Memory " adress available on the MO601 I/O module

**E) RS485 port : CHARACTERISTICS AND WIRING**

- a) RS485 type
- b) RS485 terminal
- c) Wiring

**F) MODBUS FUNCTIONS ( slave) :****Basic reminders :**

- a) Decimal hexadecimal and binary codes
- b) Building messages
- c) N ° of slave / Function / Address / data / CRC16-calculation
- d) MODBUS RTU available frames :

▪ Reading of 1 to N bits	Reading of 1 to N words	
▪ Writing of 1 bit	Writing of 1 word	fast Reading of 8 bits
▪ Writing of N consecutive bits	Writing of N consecutive words	
▪ Error message	general distribution	

**G) FRAMES EXAMPLES**

- b) Reading of bits
- c) Reading of N words
- d) Writing of a bit
- e) Writing of a word
- f) Rapid reading of N bits
- g) Writing of N bit
- h) Writing of N words
- i) Error messages
- j) Slave number "0"
- k) Maintenance

**H) MS DOS HELP MEMORY**

## A) WIRING FEATURES AND COMPONENTS

### 1 °) The various bus module available

a) Alarm annunciator panel J3000.

This panel can be fitted with a RS 485 bus link (2 or 4 wires). It becomes an intelligent sub station linked to a PLC and allow to realize an installation multitasks running and degraded Mode.

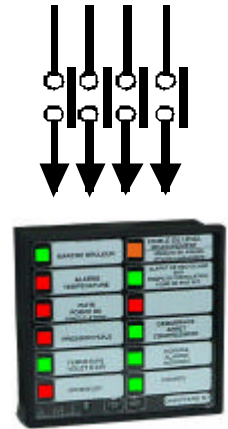
Its function is to treat local alarms with indication. A PLC or supervision can keep the alarm In the J3000 event buffer. This panel is totally self running, autonomous.

It is the only one still running in case of bus failure.

Furthermore, a group of J3000 is working separately ( each panel take care of its own channels). The PLC can keep back information, already treated by the J3000, (status, alarmhistoric) stored in the panel.

The PLC can send a visual and sonorous information to a distant operator by the activation of an input on bus. This information can come from the PLC itself or from other modules linked on the same network.

For other features and wiring diagram see J3000 leaflet.



b) Indicator panel J2000 RS 485 and J2x00 RS 485

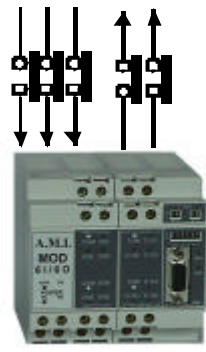
These panels get respectively 12 and 24 channels with simple indication coming from the bus. The treatment is processing by a PLC or a bus administrator G0700.

To simplify this leaflet, we only speak about J2000 RS 485 (excepted in case of variation

c) M0601 I/O RS 485 modules

They are available in 6I/6O, 12 I, 12 O, 24 I, 24 O. These economic modules keep local informations (free voltage contact) and give local free contact or remote control from G0700 bus administrator or (or PLC).

There are no local indication on module, no treatment in it. They only code or decode information to/from RS 485 bus. In case of bus fault, no back up information can be lost; They must be supplied with 24 Vcc.



### 2°) Different solutions :

a) Centralization of simple alarms

An alarm system can be made of J3000 panels. It is a substation ( for the J3000 functionalities see its leaflet)

b) Sub station linked by bus to automaton or supervision

J3000 using RS 485 Modbus Jbus protocole, it is possible to link one or several sub stations, made each one of one or more panels with a supervision or a PLC. You can use on the same network J2000 RS 485 and I/O modules. You can keep out the event buffer in each J3000 by creating a program into PLC or supervision. ( keep backs alarms, sending alarms to J3000, printing messages, remote control).

A.M.I developed a bus administrator named G0700 which already integrates all these features. For each component see specific leaflet.

- c) Sub stations linked by bus to G0700 administrator : different sub stations can be also connected to Bus administrator G0700 which integrates the main functions of management :
- Bus management
  - Date and type printing of events following the current on printer
  - Information report towards panels or output modules in local sub station  
The post of maintenance (or control room) can be made of J2000RS indicator panels or J3000 on which will be grouped various local alarms.  
Software supplied with G0700 assures :
  - The parameter setting of the administrator G0700 (parameter setting of panels and modules I/O, parameter setting of texts and reports).
  - The functions of maintenance
  - When the parameter setting of the administrator are done, the PC becomes useless.

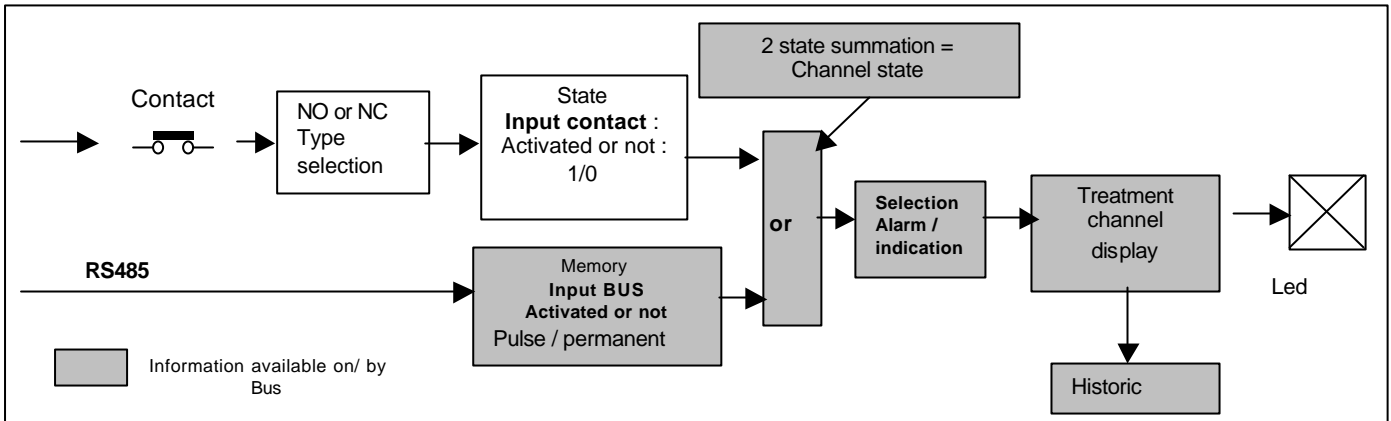
d) Sub stations linked by bus to administrator G0700 with PC / screen.

MULTITRANS II software allow to display on PC screen various alarms and events appeared on the installation. The operator will directly be able to reset on the keyboard. A "parallel" type printer can be connected to the PC moreover these linked to G0700.

**B) J3000 RS485**

**1°) J3000 functions**

a) Principle :



In standard J3000 is an alarm annunciator with 12 inputs " alarm" type . These alarms are treated according to the parameter setting of switches situated behind the J3000 panel. Result will be activation or not of the leds on fascia , the buzzer and the " alarm sound " relay, the "synthesis" relay and 12 outputs.

J3000 is an autonomous alarm automaton. It will assure the service in case of difficulties of dialogue with his supervisor.

The normal treatment from inputs contacts is :

- Change of state " Inputs contact ".
- Control of the sense(direction) + temporization of validation of the input "contact" .
- If OK, validation of the " input contact " memory
- Control of the parameter setting of the channel alarm or indication
- Treatment displaying the channel

Several functions have been added to it :

- A RS485 " Modbus / Jbus " connection
- The memorization of the setting and transmission parameters
- A bus activity control
- An internal counter "to date" events
- A memory buffer for event or "background historic"

The RS485 allows a dialogue for :

- a transfer of the panel state to a PLC or a supervision ,
- a centralization of events stored in the " historic" buffer,
- an activation by the bus of the panel's inputs,

As a consequence :

A supervisor can get back the following information by the RS485 :

- the channel state
- the state of every lighting leds
- the state of the BUS memory
- the state of the historic buffer

b) Activation of the inputs by the bus :

J3000 processing after an activated input :

An input can be activated by a contact or the bus

"contact" input :

Input « Contact »	« delay »	NO /NC	State of « input contact »	Way
If changing	= if > delay	Forwards input	=Valid Input	Validation

So the way will be displayed in alarm or indication according to its setting parameters. (see the J3000's leaflet) until the reset of the operator. It will delet if the way returned in normal state. Memory " input contact " is done by the contact remaining present or not.

On "bus" input :

« BUS »input if reception	Memory« BUS »input = Validation of input	Channel Valided
------------------------------	---	--------------------

It is possible to force immediately (without delay) the memory of the BUS input of every channel, either in indicator, or in alarm. The way will be then (without delay) displayed at once in alarm or by according to its parameter setting until the reset of the operator. It will be able to delet only ,if memory " input BUS " is cancelled. (Reception of a of disappearance order). This forcing can be pulse or permanent . This is equivalent to 2 contacts in // on the input. (One OR) .The sequences remain valid.

Also on a channel in indicator setting : if the input contact is confirmed and if an order of extinction by the busis received, the indicator led light will rest until disappearance of the contact input.

Received order by bus can be "permanent" or " pulse " type.

- Permanent : BUS order activates alarm without temporization. A new order order will be necessary to reset it (after Reset). (It corresponds to an alarm contact maintained on the input , the disappearance of the alarm making later).
  - Pulse : BUS order activates alarm without temporization. A acquit on the panel will erase indication (by taking into account the chosen sequence). (It corresponds to a pulse alarm contact on the input , alarm appears and disappears immediately, it is memorized until disappearance).
  - Case of the light : it receives a permanent order for the ignition then another order for the extinction.
- On a channel in alarm, " pulse " fonction will be realized by a validation order different from a desactivation order.

« BUS » input	Memory« BUS » input	Chanel
If reception P	= Valid Input P	Validation to acquit + reception disparition
If reception desactiveted	= not valid input D	Extinction after acquit.
If reception I	= valid input so not valid	Extinction after acquit

The panel pocesses its sequences with 2 contacts in // on every input. (Order coming from the bus is considered as an input without delay.) to erase a fault, it is necessary that these 2 orders have disappeared).

Also on a channel indicator light set : if the input contact is confirmed and if an order of extinction by the bus is received, the indicator led light rest lit until disappearance of the contact input.

Contact input state	Bus input Memory	Channel state Memory	Channel
Present		Confirmed	Confirmed until receipt = return normal
	= Confirmed entrance(entry) P	Confirmed	Confirmed until receipt + reception disappearance
Present	= Confirmed entrance(entry) P	Confirmed	Confirmed until receipt + reception disappearance + return to normal
	= Confirmed entrance(entry) then	Confirmed then not confirmed	Confirmed until receipt

c) Additive Functions:

The memorization of the transmission and configuration RS485 parameters : Possibility to change parameters of some functions by the bus and by the two buttons on fascia. These functions are stored in non volatile memory EEPROM.

- Control or not of the bus activity as well as tempo duration.
- Characteristics of the transmission. (See picture(table) annex).

J3000 is delivered with standard "factory" parameters.

**Slave N°1, speed : 9600 bauds, without parity, 8 bits of data, 1 bits of stop. Bus delay =0 min**

It is possible to reload these parameters by bus or by the buttons on fascia.

The activity bus control : Allows to check permanently the link from the panel to the network and that PLC Is still active. This security is reactivated, each time, when panel see informations on bus. It's possible to cancel This control when0 min delay is set. Other delays are available such as : 1 min, 5 min, 10 min ( set by bus). In case of alarm type " adressng bus dellay, the circular led of the fascia is activated. It is processed like an alarm channel, blinking green + alarm output relay with receipt.

**The internal meter:** A meter dates every event. This meter increments every 5ms from 0 to 5248s equal to 87mn+28s. When this value is reached, the meter is set back automatically to zero and starts again. This "meter" value is stored with every event in the buffer "historic" . It will classify events coming from different panels. (This classification should be realized by the PLC master (supervision).

Possibility to set back this meter to "0" by bus. Panels recognize slave Nbr " 0 " allowing a general broadcasting of an order sent by the master automaton. This possibility can be used to realize a general "set to 0" of several panels at the same time.

**The event buffer:** An event buffer memorize the last 64 events arising on the panel : appearance of alarms with the mode of posting facade, the receipt operator, and disappearance. The buffer is " FIFO " type , not memorized. (A break of the supply set to zero this buffer ).

Stored information include: the number of stored events, Channel and panel number, the type of installed item, the parameter setting type of the channel, the fascia display type, the meter value.

## **2°) Display and modification of the setting parameter on J3000 :**

### **Integrated from the V1 .25E version**

A push on the fascia " test " button during more than 10 s allows to show the panel configuration. Each of the various parameters will be displayed, one after the other on the led of the fascia with a particular code. It is possible to reload "factory" configuration.

Configuration "factory" is :

**Slave N°1, speed : 9600 bauds, without parity, 8 bits of data, 1 bits of stop. Tempo bus=0mm**

The led of the left column will show the step of sequence. The leds of the right column show the value of the information concerning the step of sequence displayed (for some parameters, values will be shown by means of 2 columns).

**During the sequence, the operator inactivity of 5s stops the sequence and put back the panel in normal running.**

**Any activity on the BUS will put back basic delay of 10s to 0 and will forbid passage in display mode.**

**It means that it is necessary to disconnect the panel from the bus to make any modification.**

### **Modification of the setting parameter :**

The operator presses on the PB test led, in a time of 10s. Test led is done and a blinking of all the leds appears :

- If the operator press off the button and push it again, the sequence of display begins.
- Otherwise, the panel return to normal functioning.

During the parameters display, the button " Reset " allows to modify setting. This setting will become actual in the following step. (The operator has to confirm the following step with the button "test" before the end of temporization of 5s otherwise the panel resumes its normal functioning and modification is not memorized).

N°	Pas	Step
N°1	Speeds	The led N°1 indicates the first program step : speed. Led N ° 7,8 , 9 , 10, 11 display in fix the config. ( 7 =1200 , 8 = 2400 , 9= 4800 , 10 = 9600 , 11=19200). Possibility to modify setting speed by the " reset " button.
N°2	Nbr of bits	The operator presses again on the "led test " PB . The second led N°2 indicate the 2 nd program step : Nbr of data bits. The leds N°7 , 8, indicate in fix the config. ( 8= 8 bits). <b>This parameter is not modifiable.</b> It is fixed to 8 bits .
N°3	Stop bits	The operator presses again on the "led test" PB. The third led N°3 indicate the 3 rd program step : Nber of stop bits. The leds N°7,8, indicate the config. ( 7=1 , 7+8 = 2). Possibility to change setting by the " reset " button .
N°4	Parity	The operator presses again on the "led test" PB. The led N°4 indicate the 4 -th program step . The leds N°7,8 , 9 indicate in fix the config. ( 7 =without). <b>This parameter is not modifiable.</b>
N°5	Slave Nbr	The operator presses again on the "led test" PB. The led N°5 indicate the 5 -th program step : Slave nbr. led N ° 7 to 12, displayed in fix the config. of dozens (v7+v8+v9 =30). After a press, the led 1 for 9 displayed in blinking units. (V1+v2+v3+V4=4 so 30+4=34). Possibility to change seting by the "led test" PB.
N°6	Bus Control delay	The operator presses again on the "led test" PB. The led N°6 indicate the 6 -th program step in fix (v7=sans , v8=1mn , v9=5mn, v10=10mn). Possibility to change setting by the "led test" PB.
N°7	Basic factory settings	The operator presses again on the "led test" PB. All the leds are blinking SLOWLY. -A press on the "test" button put the panel out of the program. - During the display blinking slowly, a press on the " reset " button put display in fast blinking. During this display , a press on the "test" button reloads "factory" configuration. If the operator does not make anything and waits for the end of 5s delay, the panel will return to normal mode without taking into account "factory" setting. Reminder :Configuration "factory" : Slave N°1, speed : 9600 bauds, without parity, 8 bits of data, 1 stop bit. Bus delay = 0 min



**3°) Memory address available on J3000 RS485**

a) Reading of led state

Function code 3	01 (H)/ 01	Word	R	State of led Channel 1	<b>0 = Lit off</b> <b>1 = Lit</b> <b>2 = Flash blinking</b> <b>3 = Quick blinking</b> <b>4 = Slow blinking</b>
	02 (H)/ 02	Word	R	State of led Ch 2	
	.....	.....	.....	.....	
	0c (H)/ 12	Word	R	State of led Ch 12	

Explanation :

It is possible to read the group of 12 addresses in only one frame or to read the addresses you wish.  
 Attention : during a reading, it is imperative that the frame of reading does not exceed 0C H (12) address.

b) Reading / Writing in the bus input memory :

Function Code 3 and 6	11 (H) / 17	Word	W/R	State of memory bus input Ch 1	<b>0 = Not valided memory</b> <b>2 = Valided memory pulse</b> <b>3 = Permanent validedmemory</b>
	12 (H) / 18	Word	W/R	State of memory bus input Ch2	
	.....	.....	.....	.....	
	1c (H) / 28	Word	W/R	State of memory bus input Ch 12	

Explanation :

READING :

It is possible to read the group of 12 addresses in only one frame or read the addresses you wish.

Attention : during a reading, it is imperative that the frame of reading does not exceed 1C H (28) address and does not begin before the 11 ( H ) / 17 address.

WRITING :

It is possible to activate Bus memory in pulse or in permanent way. The writing is made channel by channel :

Pulse : memory is activated but it is not necessary to send a code to deactivate it. This function is egal to the change of contact state which return to the normal.

Permanent : memory is activated, it is necessary to send a new frame to deactivate it.

c) Reading of the channel state (contact input state, bus input state)

Function code 3	21 (H) / 33	Word	R	State of Ch 1	<b>0 =desactivated</b> <b>1 =activated</b>
	22 (H) /34	Word	R	State of Ch 2	
	.....	.....	.....	.....	
	2c (H) / 44	Word	R	State of Ch 12	

Explanation :

READING :

It is possible to read the group of 12 addresses in only one frame or to read a reading the addresses you wish. Attention : during a reading, it is imperative that the frame of reading does not exceed 2C ( H ) / 44 address and do not begin before the 21 ( H ) / 33 address.

a) Reading the Channel setting parameter depend on the S22 and S23 switch position behind)

Function code 3	31 (H)/ 49	Word	R	Parameter of Ch 1	<b>0 = Channel in indicator</b> <b>1 = Channel in alarm</b>
	32 (H)/ 50	Word	R	Parameter of Ch 2	
	.....	.....	.....	.....	
	3c (H)/ 60	Word	R	Parameter of Ch 12	

Explanation :

READING :

It is possible to read the group of 12 addresses in only one frame or to read the addresses you wish. Attention : during a reading, it is imperative that the frame of reading does not exceed 3C H (60) address and does not begin before the 31H (49) address.

b) Reading / Writing on the KL alarm relay :

Function code 3 et 6	40 (H) / 64	Word	W/L	Set to 0/1 by the Bus / "Reset" PB on fascia or the outside input stop Alarm sound and contact. In case of activation of the KL relay by the bus, the operator can reset it by the button on fascia or by the stop KL input and vice versa.	<i>1 = active</i> <i>0 = desactive</i>
-------------------------	-------------	------	-----	--	---

Explanation :READING :

It is possible to read the state of the KL relay in a single word.

WRITING :

It is possible to activate the KL relay by sending a writing writing to 40 H (64) address ( 1 Only WORD)

c) Writing for the Led Test:

Function code 5 et 6	42 (H) / 66	Byte &Word	W	Generate a leds test of x seconds on the panel	1 = switch on "Test leds"
-------------------------	-------------	---------------	---	--	---------------------------

Explanation :WRITING :

It is possible to activate a "led test" by sending a writing frame to address 42 H (66) address (In Word or in Byte)

d) Writing « reset » on the panel:

Function code 6	43 (H) / 67	Word	W	Reset the panel ; Same function a rear clamp but the function code will be done whatever the Reset step on the panel. ( A code 2 provokes stop alarm and stop blinking if alarm sound is there)	1 = AKL, 2=ACL, 3=EFF AKL = Stop alarm ACL = Stop blinking EFF = Reset
--------------------	-------------	------	---	---	---

Explanation :WRITING :

It is possible to activate a "reset" on the panel by sending a writing frame to 43 H (67) ( 1 only word).

e) Reading Reset Opérateur (not available function) :

44 (H) / 68	Word	R	Return a code indicating the state in the sequence : 1 = AKL ok, 2=ACL ok, 3=Eff ok
-------------	------	---	--

Temporarily, waiting for this function, to verify the reset of an operator, it is possible :

- To verify if a led is blinking. (So "ok", the operator did not reset).
- To look at the presence of " the receipt operator " in the buffer event.

Explanation :READING :

It 's possible to read the « reset » state with only one word.

f) Standard parameter :

Function code 3 et 6	Transmission speed	80 (H)/ 128	Words	W/ R	1200 bauds = F0 H (240) 2400 bauds = F8 H (248) 4800 bauds = FC H (252) 9600 bauds = FE H (254) 19200 bauds = FF H (255)
	Nber of stop bit	81 (H)/ 129	Words	W/ R	1 or 2
	Number slave	82 (H) /130	Words	W/ R	1 at 64
	Reception fault RS 485 delay	83 (H) / 131	Words	W/ R	0 = Inactive 1 = 1 minute 2 = 5 minutes 3 = 10 minutes
	Type Material	84 (H) / 132	Words	R	A code for the supervision, specifying the item set on the bus (here, 1=J3000 , 2=J2000RS485).

Explanation :READING :

The reading on 80 (H) to 84 (H) address can be done word after word .

WRITING :

The writing will be made word after words from the address 80 ( H) to 84 ( H) .

“Factory” set up : It can be done either by the sequence on the fascia push buttons or by the writing at the various addresses from 80 ( H ) in 84 ( H ) (9600 Bauds, 1 Stop, Slave 1, delay 0)

g) **Maintenance** (Funtion codes available) :

Function code 3	Slave Nbr 65	82 (H) / 130	Words	R	Useful with present 1 only slave on the BUS which will return the number of panel.
-----------------	--------------	--------------	-------	---	--

**Explanation :**

**READING :**

Reading the address 82 ( H ) /130 with slave number 65 allows to get back the number of the slave (1 - 64).

h) **Functioning of the "events" buffer :**

Function code 3 and 6	Current state of the panel, Counts of events	90 (H) / 144 91 (H) / 145	words	R	Up to 64 events. It is contained in “ panel state” address 90 and 91. <ul style="list-style-type: none"> <li>Word 90 ( H ) : Reserve ( 8 bits) + Nbr of events to be read ( 8 bits) = 1 Word.</li> <li>Word 91 ( H ) : Number of the panel ( 8 bits) + Type of material ( 8 bits) = 1 Word.</li> </ul> To see frame below
	Read a part of the event buffer	92 (H) à 111(H) (146 à 274)	2 words	R	To read from 1 to x Up to 64 events egal to 128 words. => 1-st word : Channel ( 4 bits) + Channel parameter type (alarm/indicator) +( Type of display type ( 4 bits) Alarm / indicator) ( 4 bits) + internal meter index ( 4 bits) => 2-nd word : internal meter value ( 16 bits). " The operator reset " makes part of for information stored in event buffer.
	Event buffer reset	120H(288)	word	R	Number of events to erase (1 - 64). Erase n first events of the pile.
	Meter reset	121 H(289)	word	R	Allows to reset the counter. Only value 0 is accepted.

90 (H)	0000	0000	0000	0000
	Reserved	Reserved	Nb events to read	Nb events to read
91 (H)	0000	0000	0000	0000
	Panel number	Panel number	Item type	Item type

Visualisation of an event on 2 Words (Every information takes 2 words = 128 words for 64 events ) :

92 (H) à 111 (H)				
Word 1	0000	0000	0000	0000
	Ch number	Ch parameter type (alarm/indicator)	Display type	Counter value
Word 2	0000	0000	0000	0000
	Counter value	Counter value	Counter value	Counter value

**Explanation :**

**N°of channels :**

- 1 - 12 = for input channel
- F (16) = for inputs PB « stop KL » / « stop CL / EFF (Erase)

**Channel parameter setting :**

- 1 = Channel in alarm
- 0 = Channel in indicator way

**Display type :**

If channel number from 1 to 12

- 0 = Light off
- 1 = Lit fix
- 2 = Blinking Flash
- 3 = Blinking fast
- 4 = Blinking slowly

If channel number F (16)

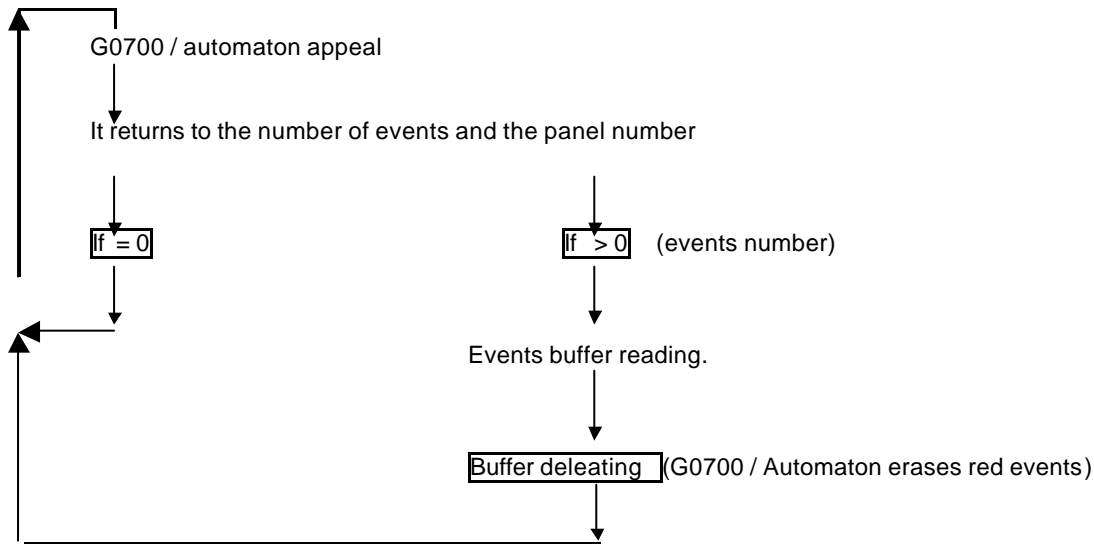
- 1 = PB stop alarm (KL relay)
- 2 = PB stop blinking (CL)
- 3 = PB Reset

**Counter value :**

Internal meter allowing a classification of events between various panels.  
(Increase meter 5 ms)

**4°) Functioning of the events buffer :**

- To read the nb of present events (the nb is contained in " panel's state " in 90 (H) adress.
- Reading of the event buffer if the number of event contained in the address 90 ( H ) is different from zero
- After the reading erase the events from 1 to X (Adress 120 + nbr of events to erase (1à64)). J3000 erases the first nbr of the pile).



**5°) Software added on J3000RS485**

Version 1.25 : addition of the display / modification of the transmission parameters by the fascia's buttons TEST and RESET

**C) J2000 RS485 and J2x00 RS485**  
**Indicator panel**

J2000RS485 is a panel with 12 channels.  
 J2x00RS485 is a panel with 24 channels.

On these 3 following references, only the fascia change :

J2100RS485 is a 24 channels (one by label)  
 J2200RS485 is a 12 channels (two by label, example: ON-OFF or RUN-FAULT)  
 J2300RS485 is a 8 channels (tree by label, example: ON-OFF-Fault)

J2000 and J2x00 get the same functions apart from the channel's number available.  
 For the next explanations, we will only speak about J2000, excepted where a difference is important.

**1°) Function of the J2000RS485 or J2x00RS485 :**

a) **Principle**

“Power led” on fascia : Tricolour, this led indicates :

- green : normal functioning of the panel.
- orange : in a quick time, indicates a fleeting transmission error .
- red : indicates an anomaly on panel, then " Watch dog " relay is one, no activity on bus meaning bus is out of order.

“Test " led button (on fascia) (this button is active only on panel) :

- a permanent press light in permanent all the leds .
- according to the panel's setting , the Test led PB can reset the “user relay” and the optional internal buzzer.

If the user relay is activated and the panel set in “reset allowed”, a press on the “Test” button reset user relay. Reset operator memory shift from 0 to 1 value during 30 seconds. Led test will not be done; Tst button is not activated on blinking led mode. It is necessary to send a frame to set back the led after Reset operation.

“Aux” push button (on fascia) :

- This PB is link to 1A and 2A clamp on rear side; It can use as a free voltage contact up to 5A/250 Vca to give a remote information.

b) **Visualization of the setting on the J2000 RS 485 fascia :**

The operator press 3 times rapidly and after press permanent on the led “test “ button , in a time of 3s ( 3 short push + 1 long).

- \* If the sequence of press is not respected, only the led test regular is realized.
- \* If the sequence is respected, display of 1step (the left led Number 1).

=> Several actions on the button allow to display the configuration of the panel by using leds of front face (and, if necessary, to reload the " factory setting "). The " factory setting" is :

**Slave number 1, speed : 9600 bauds, without parity, 8 bits of data, 1 bit of stop, acknowledge authorization user relay.**

- The leds of the left column will display the step of sequence. The leds of the right column will display the value of the information concerning the state of sequence displayed. During the sequence, all inactivity of 6s stops the sequence and provokes a return to normal. All reception of framework on the RS485 port (therefore destined to the considered panel) provokes a general lamp test of 2s and a return to normal, whatever the step of the sequence program.

Step number	Step	
1	Speed	Led 1 is on, it indicates 1first step of program :speed Leds 13,14, 15, 16, 17 display in fix light the setting. (13 =1200, 14 = 2400, 15 =4800, 16 = 9600, 17= 19200).
2	Number of bits	The operator pushes a new time on the lamp test button .Led 2 lights fix indicating 2sd step of program : Number of data bits. The leds 13 to 20, display in fix light the setting (20 =8 bits).
3	Stop bits	The operator pushes a new time on the led test button .Led 3 lights fix indicating step 3 of program : Stop bits. The leds 13,14 , display in fix light some the setting: (13 =1, 13+14 = 2).
4	Parity	The operator pushes a new time on the led test button. Led 4 lights in fix indicating step 4 of

		program : Parity. The leds 13,14 ,15 display in fix light some the setting: (13 =sans, 14 = pair, 15=impaire).
5	Slave number	The operator pushes a new time on the led test button. Led 5 lights in fix indicating step 5 of program : Slave number. The leds 7 to 12 display in fix light and indicate the Tens => (v7+v8 =20). Leds 13 to 19 display in fix some units. (v13+v14+v15=3 is 20+3=23).(other example : if , only led 13 is lighted = 0+1=1)
6	Synchronization	Led 6 lights by permanent indicating step 6 of program : synchronisation . Leds 13 and 14 display fixes some the setting. (13 in permanent = synchronization master, 14 in permanent = receiver of synchronization, slave).
7	Bus control	The operator pushes a new time on the led test button. Led 7 lights by permanent indicating step 7 of program : Bus control (v13=No control, v14=1mn,v15=5mn, v16=10mn).
8	Acknowledge authorized	The operator pushes a new time on the lamp test button. Led 8 lights by permanent indicating step 8 of program : Acknowledge authorized. (v13= no acknowledge, v14= with acknowledge)
9	Factory setting	The operator pushes a new time on the lamp test button. Led 9 lights blinking and indicate step 9 of program : Factory setting. Inactivity of 6s stops the sequence and provokes a return to the normal. =>If the operator makes a long "push" on the lamp test , all leds on the panel blink. =>If the operator remains pushed while the panel blinks: - After 10s, the panel recharges its " factory configuration" and returns to normal . - If it pushes less 10s, the panel returns to normal without changing its configuration.

**2°) Memory Address available :**

Following address are reachable on J2000 and J2x00 RS485 excepted for channel 13 to 24 non existent on J2000 (only 12 channels).

**a) Reading/ Writing of the various states:**

Function code 3 / 6	01(H) / 01	Words	Read / write	Configuration on the led 1	0 = Off 1 = Lighted 2 = flash Indicator 3 = rapid Indicator 4 = slow Indicator
	02 (H) / 02	Words	Read / write	Configuration on the led 2	
	18 (H) / 24	Words	Read / write	Configuration on the led 24	
	19 (H) / 25	Words	Read / write	Configuration on leds 1 to 5 (3 bits by LED to code 0 to 4) the bit of weight strong is not used.	
	1A (H) / 26	Words	Read / write	Configuration on leds 6 to 10 (3 bits by LED to code 0 to 4) the bit of weight strong is not used.	
	1B (H) / 27	Words	Read / write	Configuration on leds 11 to 15 (3 bits by LED to code 0 to 4) the bit of weight strong is not used.	
	1C (H) / 28	Words	Read / write	Configuration on leds 16 to 20 (3 bits by LED to code 0 to 4) the bit of weight strong is not used.	
	1D (H) / 29	Words	Read / write	Configuration on leds 21 to 24 (3 bits by LED to code 0 to 4) the bit of weight strong is not used.	

Function code 1,2,3 / 6	Horn relay	40 (H) / 64	Bit/Word	Read / write	State of the user relay 1 = Work 0 = Reset
Function code 1,2,3 / 6	Synchronization	41 (H) / 65	Bit/Word	Read / write	1 = emitter 0 = slave (receive)
Function code 5 / 6	Lampe test	42 (H) / 66	Bit/Word	Write	Generates a Lampe test of 1 second on the panneau 1 = lampe test (255)
Function code 1,2,3,5 / 6	Acknowledge Authorization	43 (H) / 67	Bit/Word	Read / write	1 = with authorized acknowledge 0 = without acknowledge
Function code 1,2,3	Acknowledge Memory	44 (H) / 68	Bit/Word	Read / write	Internal memory giving information for the user relay or buzzer reset/or not by the operator; If the user relay is activated and the panel set in "acknowledgement authorized" a push on "led test" button reset user relay. Acknowledge operator memory shift

					from to 1 during 30 s. Led test could not be done.
Function code 5		64 (H) / 100 to C3(H) / 195	Bit	write	Configuration of a led by 4 bits block

**b) Standard setting:**

Function code 3 / 6	speed	80 (H) / 128	Words	Read / write	1200 bauds = F0 H (240) 2400 bauds = F8 H (248) 4800 bauds = FC H (252) 9600 bauds = FE H (254) 19200 bauds = FF H (255)
	Stop bits	81(H) / 129	Words	Read / write	1 or 2
	Slave number	82(H) / 130	Words	Read / write	( 1 to 64)
	Control bus delay	83(H) / 131	Words	Read / write	( 0 = Inactive ) ( 1 = 1 minute ) ( 2 = 5 minutes ) ( 3 = 10 minutes )
Function code 3 / 6	Item type	84(H) / 132	Words	Read	indicates a code to the supervisor, specifying equipment link on the bus (here, 1=J3000, 2=J2000RS485)
				write	It is necessary to set all values " factory " step by step ( 9600 bauds = FE (H) / 254), without parity, 8 bits data, 1 bit of stop , bus delay =0 Possibility to use the setting by buttons of fascia

The only event stocked in the RS485 J2000 is " the operator acknowledgement ". See to the § " Memory acknowledgement ".

**3°) Different versions of the J2000 RS 485 program :**

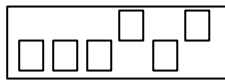
- Version 1 : 2 bits of stop, 1200 bauds,
- Version 2.04 : in standard setting, the panel is delivered in configuration : 1 bit of stop, 9600 bauds.
- Version 2.06 : different modification about the software of the 2 wires communication..

**D) MO601 I / O RS485 MODULE**

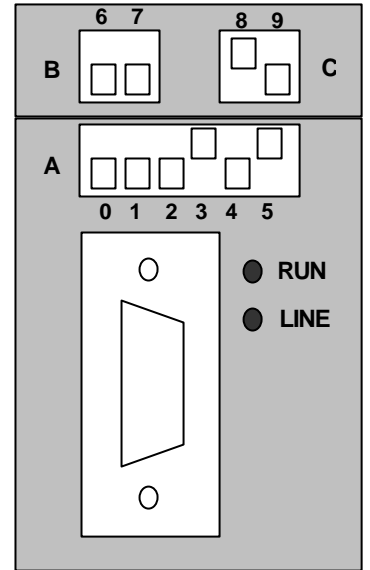
**1°) Function of I / O module**

- Parameter of transmission :  
 -To select N ° of slave on Dip switch A.  
 S0 => bit of Strong weight    S5 => bit of weak weight

Example : slave 5 => 000101 =>



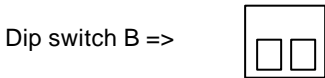
Dip switch A	
N° Slave	Switches 0 1 2 3 4 5
1	0 0 0 0 0 1
2	0 0 0 0 1 0
3	0 0 0 0 1 1
4	0 0 0 1 0 0
5	0 0 0 1 0 1
10	0 0 1 0 1 0
15	0 0 1 1 1 1
20	0 1 0 1 0 0
30	0 1 1 1 1 0
60	1 1 1 1 0 0
63	1 1 1 1 1 1



Attention, change slave Nbr only on a device out of voltage. Otherwise, new selection Will be not taken into account

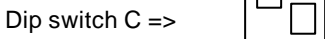
- To select the size of transmission on Dip switch B and C:  
 For an use with the **G0700** administrator and the other **A.M.I** modules,  
 Always use : 8 bits , without parity.

Example : speed 9600 bds => B6=0    B7=0/1    C8=1    C9=0

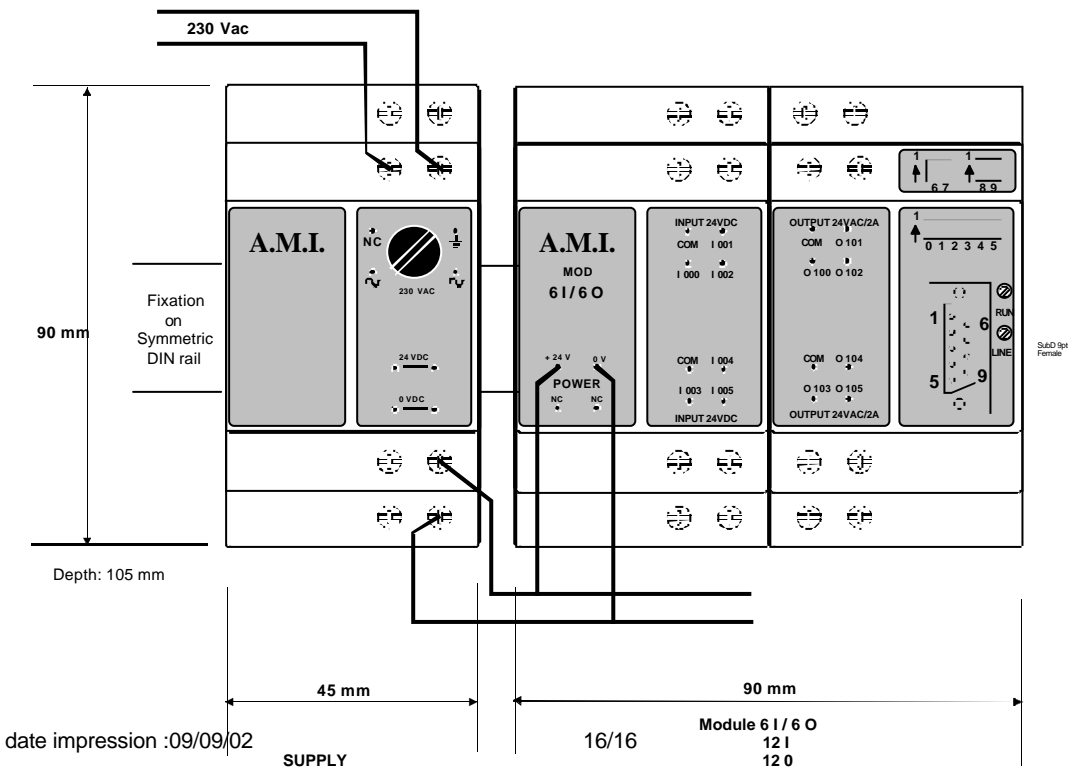


Dip switch B		
B6	B7	Parity
0	0/1	No
1	1	Pair
	0	impair

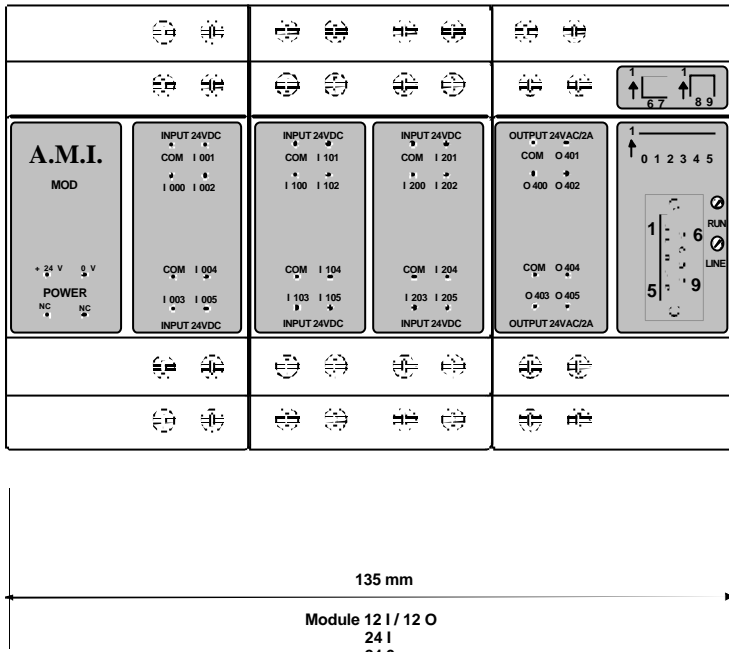
Dip switch C		
C8	C9	Speed
0	0	38400 bds
0	1	19200 bds
1	0	9600 bds
1	1	4800 bds



**2°) Dimensions and connections**



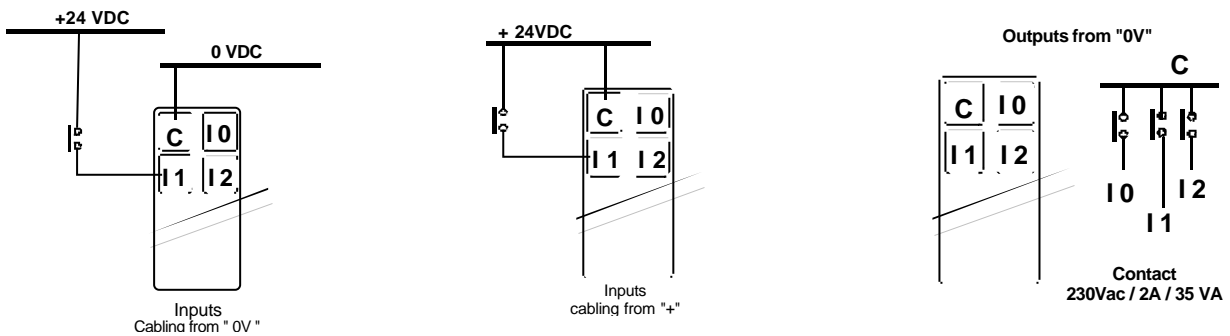




4°) Memory adress available on the MO601 I/O module

Module type	Block Nbr	Channel nbr
6 I / 6 O	1	I => I 000 à I 005 O => O 100 à O 105
	2	
12 I	1	I => I 000 à I 005
	2	I => I 100 à I 105
12 O	1	O => O 000 à O 005 O => O 100 à O 105
	2	
12 I / 12 O	1	I => I 000 à I 005
	2	I => I 100 à I 105
	3	O => O 200 à O 205
	4	O => O 300 à O 305
24 I	1	I => I 000 à I 005
	2	I => I 100 à I 105
	3	I => I 200 à I 205
	4	I => I 300 à I 305
24 O	1	O => O 000 à O 005 O => O 100 à O 105
	2	O => O 200 à O 205
	3	O => O 300 à O 305
	4	O => O 400 à O 405

3°) Channel wiring :



## E) RS 485 PORT, CHARACTERISTICS AND WIRING

**a) RS 485 type :**

- Half Duplex Interface type (reception and emission are not simultaneous)
- Isolated by opto - coupling.
- 2 wires or 4 wires. Possibility to connect in 1 paire emission / reception or 1 paire emission + 1 paire reception.(Selection by external connexions on the terminals ).
- Speed of transmission (1200,2400,4800, 9600, 19200 bauds), mode without parity, transmission on 8 bits, 1 stop bits,
- Number of slave from 1to 64, setting by the RS485 port.
- Resistors of end line of 150 ohms of the RS485 link are out of the interface.

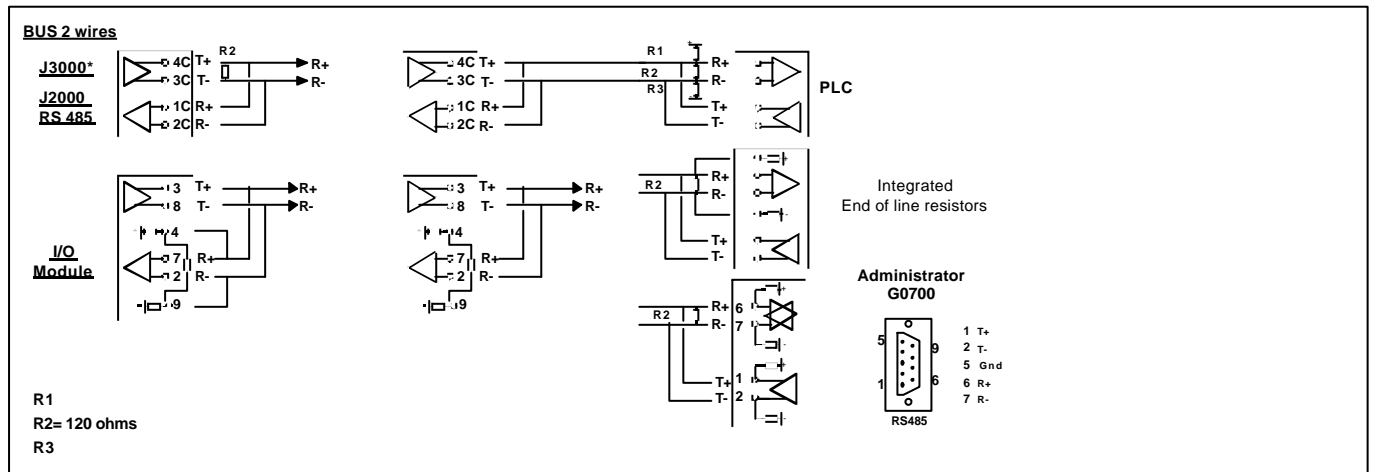
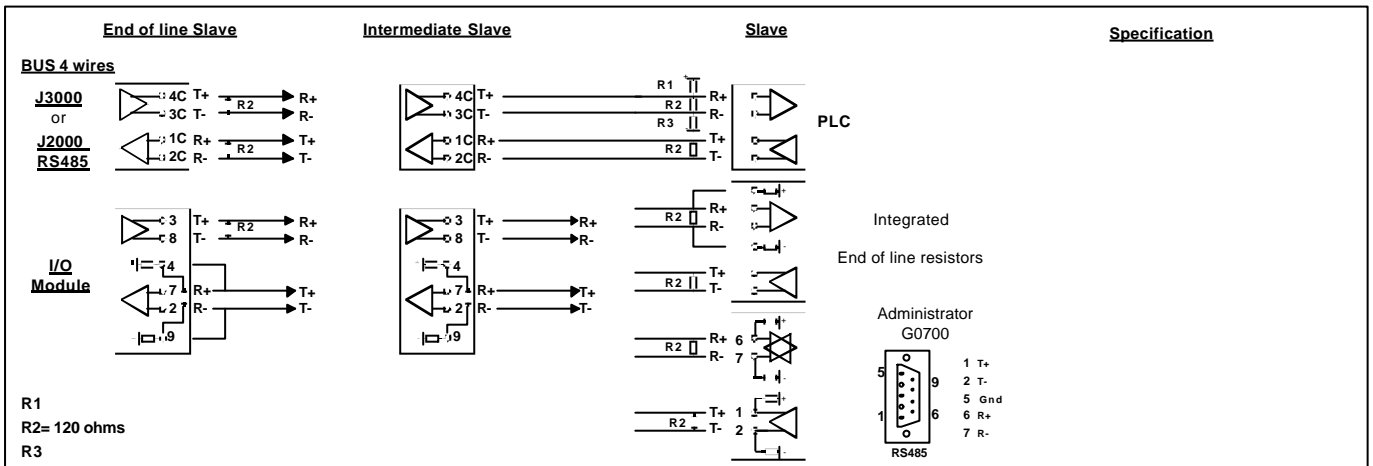
**b) RS 485 terminal : 4 plug in terminals:**

- 1C/2C : connection RD+/RD -
- 3C/4C : connection TD -/TD+
- mass : The mass has to be connected to the plating of the BUS cable and to the general chassis.

" green " led: visualize the passage of framework in Emission coming from the panel.

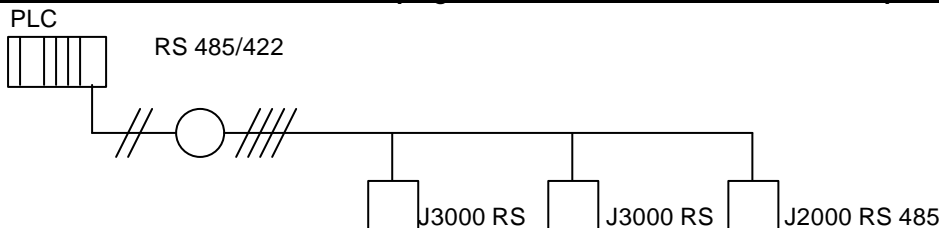
" red " led: visualize the passage of framework in Reception coming from the BUS.

**c) Wiring :**



If the master is a G0700 of AML, if the end of line slave is an I/O module , connections between 4 and 7 as well as 2 and 9 are useless.

**\*if J3000 is used in 2 wires network , plug it behind 2/4 wires converter for example ADAM 4510 (RS 485/422).**



## F) MODBUS FUNCTIONS (slave)

**Basics reminders :**

a) Decimal, hexadecimal and binary value codes :  
 An information can be coded of different manners :

Decimal	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Hexadecimal	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Binary	1	10	11	100	101	110	111	1000	1001	1010	1011	1100	1101	1110	1111

Decimal	16	17	20	30	31	32	47	63	175	255
Hexadecimal	10	11	14	1E	1F	20	2F	3F	AF	FF
Binary	1 0000	1 0001	1 0100	1 1110	1 1111	10 0000	1 01111	11 1111	10101111	1111 1111

Decimal	511	2815	$(2 \times 10^3) + (8 \times 10^2) + (4 \times 10^1) + (2 \times 10^0) = 2842$
Hexadecimal	1FF	AFF	B1A (H) => B (H)=11 1 (H)=1 A (H)=10 $(11 \times 16^2) + (1 \times 16^1) + (10 \times 16^0) = 2842$ $2816 + 16 + 10 = 2842$
Binary	1111 1111 1	1010 11111111	1 0 1 1 1 0 0 0 1 1 0 1 0 $(1 \times 2^{11}) + (0 \times 2^{10}) + (1 \times 2^9) + (1 \times 2^8) + (0 \times 2^7) + (0 \times 2^6) + (0 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) = 2842$ $2048 + 0 + 512 + 256 + 0 + 0 + 0 + 16 + 8 + 0 + 2 + 0 = 2842$

A value can be coded in binary, decimal or hexadecimal.

It is indicated on 16 bits : 0000 0000 0000 0000.

example : 0000 0000 1111 1111 (binary) = 255 (decimal) = FF (Hexadecimal)

or again 0000 0000 1010 1100 = 172 = AC (H) (H means : in hexa.)

This value can be on 1 or 2 bytes, (1 byte = 8 bits), what is writing : Byte 1 + Byte 2.

First part of the byte : 1010 is called WS or Weight Strong. Second part : 1100 is called ww weak weight.

The maximal value of a byte is 255. A value (decimal) superior will write :

Example : 637 =  $(2 \times 256) + 125$  => 02,125 02 is the weight Strong.

If the coding is on 2 bytes : 1111 1111 1111 1111 = 65535 = FF FF or 255, 255

Noted : In the continuation of the documentation, we will indicate values on 2 bytes in Hexadecimal or decimal.

b) Building messages:

The Interface functions according to the protocol MODBUS/JBUS

Reading :

Slave number	FUNCTION Number	ADDRESSES	Number of bits/word	CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes

Reply in bit :

Slave number	FUNCTION Number	Number of bytes read	Value 1 to value x	CRC16
1 Byte	1 Byte	1 Byte	1 Byte .....1 byte	2 bytes

The reply is made always on 1 byte. 1st bit red is put on the weak weight of the byte containing the bit. The unused bits are set to zero.

Reply in word :

Slave number	FUNCTION Number	Number of bytes read	Value 1 to value x	CRC16
1 Byte	1 Byte	1 Byte	2 Bytes .....2 bytes	2 bytes

c) Slave Number : On the bus, there is only one master and several slaves. Each slaves (J2000RS485, J3000RS, Module E/S) gets a number for identify them. This number has to be unique (2 slaves can not have the same number).

The master can be a COMPUTER, a programmable automaton, the administrator of "A.M.I. Bus" : G0700. It takes the initiative to ask questions to the slave. This one verifies that the order received is well for it (with the Number of slave), executes the order and returns a reply to the master if necessary.

Function : made on 1 byte

Function codes recognized :

- Reading of 1 to N bits : 1 or 2 ( used in the J3000)
- Reading of 1 to N words : 3 or 4
- Writing of a bit : 5

- Writing of a word : 6
  - Writing of N bits : 15
  - Writing of N words : 16 (only for the J2000RS485)
- General distribution, Words, writing : 0 All present slaves process demand , but do not reply.  
 The slave number 65 is used in maintenance to find the address of a module (in the J3000 only).  
 (address containing the number of slave).

- The interface send messages of error following if necessary.
- Error codes
  - Code unknown function : 1
  - Incorrect address : 2
  - Incorrect data : 3
  - Impossible to realize : 4
  - Others to define : 5

**Address :** It concerns the word address or the bit to read. On 2 bytes. (See each product tables).  
 It is possible to read or to write on several addresses. One indicates the address of departure, next addresses will be red or written on 2 bytes.

**Number of bits/words :** on 2 bytes. Indicates the number of addresses that have been specified previously in the message.

**Number of bytes read :** Indicates the number of bytes of reply that follow.  
 In the case where the reply is :

- ✓ in bit , it is coded on a byte . The number of bytes read correspondent to the number of data receive.
- ✓ in word, it is coded on two bytes. The number of bytes read correspondent to the number of data receive x 2 .

**Data 1 to x :** (in response). Read Values . Always on 1 byte for reply in bit. With 2 bytes for replies in word. Several replies can follow. The data number is indicated in the " Number of bytes red ". ( multiplied by 2 if data are in word )

**CRC16 - calculation :**

Xor is :

+	0	1	0	1
+	0	0	1	1
=	0	1	1	0

It calculates according to the next manner : (example with 2 bytes = 2 (H) = 2 = 00C

CRC16= FFFF (H) = 65535 =>

(CRC16) Xor (1st byte) = CRC16 Xor is "or exclusive"  
 new CRC16

1111 1111 1111 1111  
 0000 0010  
 1111 1111 1111 1101

**1st step :**

to offset the CRC16 of 1bit to the example right  
 new CRC16

1111 1111 1111 1110 1  
 0111 1111 1111 1110

If the bit outgoing is 1 (it is the case in the example), to make CRC16 Xor A001 (H)

If the bit outgoing is 0, don't do anything

A001 (H) = 40961 = 1010000000000001  
 new CRC16

1010 0000 0000 0001  
 1101 1111 1111 1111

**Step 2 :**

to offset the CRC16 , new CRC16  
 to make CRC16 Xor A001 (H) (the bit outgoing =1)  
 new CRC16

0110 1111 1111 1111 1  
 1010 0001  
 1100 1111 1111 1110

**Step 3 :**

(the bit outgoing =0 don't do anything)

0110 0111 1111 1111 0

**Step 4 :**

to offset the CRC16 , new CRC16  
 to make CRC16 Xor A001 (H) (the bit outgoing =1)  
 new CRC16

0011 0011 1111 1111 1  
 1010 0001  
 1001 0011 1111 1110

**Step 5 :**

(the bit outgoing =0 don't do anything)

0100 1001 1111 1111 0

**Step 6 :**

to offset the CRC16 , new CRC16  
 to make CRC16 Xor A001 (H) (the bit outgoing =1)  
 new CRC16

0010 0100 1111 1111 1  
 1010 0001  
 1000 0100 1111 1110

**Step 7 :**

(the bit outgoing =0 don't do anything)

0100 0010 0111 1111 0

**Step 8 :**

to offset the CRC16 , new CRC16 0010 0001 0011 1111 1  
 to make CRC16 Xor A001 (H) (the bit outgoing =1) 1010 0001  
 new CRC16 1000 0001 0011 1110

1001 0001 0011 1110 = 33086 = 813E (H)

the value of the CRC16 is indicated ww( weak weight)-WS(weight strong) => 3E,81=> 62,129 in hexadecimal.

**d) MODBUS available frames :**

**Read of 1 to N bits (expressed in decimal) (N = 1 to x) Function 1 or 2**

Slave number	FUNCTION Number	ADDRESSES of first byte	Number of bits read	CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes
01	01	00 64	00 01	252,30

example :

01 => slave 1

01 => N° of function. Here the function to use is 1.

00 64 => address of 1st bit to read (64= state relay reading on the J2000RS) Note : 00 64 (decimal) => 00 64 (H) (hexadecimal)

00 01 => number of bits (here, 1)

252 30 => result of the calculation CRC16. the value of the CRC16 is sent ww ( weak weight)-WS(weight strong) :

252,30=>252=FC 30=1E => 252,30=> FC,1E=> 1EFC=7932 (in decimal).

The reply is always made on 1 byte. 1st bit read is placed on the weak weight of the byte containing the bit. The bits unused are to zero.

reply :

Slave number	FUNCTION Number	Number of bits read	First byte	Last byte	CRC16
1 Byte	1 Byte	1 Byte	1 Byte	1 Byte	2 Bytes
01	01	01	00		81 136

01 => slave 1

01 => Number of function. Here the function to use is 1.

01 => Number of bytes read. Here 1

00 => state of the bit red (00=lecture state of the relay on the J2000RS. The relay is not activated)

Noted : 00 (decimal) => 00 (hexadecimal)

81 136 => result of the calculation CRC16

If the number of byte to read is N, the reply comprises N replies corresponding to the address X =>X=N.

**Reading from 1 to N words (expressed in decimal) (N = 1 to x) Function 3 or 4**

Slave number	FUNCTION Number	ADDRESSES of first word	Number of word read	CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes
01	03	00 08	00 01	5,200

01 => slave 1

03 => Number of function. Here the function to use is 3.

00 08 => address of the word to read (08=lecture state the led 8 on the J2000RS)

00 01 => number of words (here, 1)

5 200 => result of the calculation CRC16

reply :

Slave number	FUNCTION Number	Number of words read	First word	Last word	CRC16
1 Byte	1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes
01	03	02	00 03		248 69

01 => slave 1

03 => Number of function. Here the function to use is 3 or 4.

02 => Number of bytes read ( Noted : 2 bytes => 1 word = 2 bytes = number value x 2 )

00 03 => state of the read word (03=lecture state of the led 8 on the J2000RS. The led is rapid indicator)

248 69 => result of the calculation CRC16

**Writing of a bit : Function 5**

Slave number	FUNCTION Number	ADDRESSES of byte	Value of bits		CRC16
1 Byte	1 Byte	2 Bytes	1 Byte	1 Byte	2 Bytes
01	05	00 66	255	00	44,46

example :

01 =&gt; slave 1

05 =&gt; Number of function. Here the function to use is 5.

00 66 =&gt; address of the bit to write (66=write led test on the J2000RS/J3000)

Noted : 00 66 (decimal)=&gt; 42 H ( hexadecimal)

255 is the value of the bit (here, all bits of the byte are to 1 is 1111 1111 or FF or 255)

0 the value of all the bits of the byte are to 0 is O(H) is 0 in decimal.

00 =&gt; byte empties

44 46 =&gt; result of the calculation CRC16

reply :

Slave number	FUNCTION Number	ADDRESSES of byte	Value of bits		CRC16
1 Byte	1 Byte	2 Bytes	1 Byte	1 Byte	2 Bytes
01	05	00 66	255	00	44,46

01 =&gt; slave 1

05 =&gt;Number of function. Here the function to use is 5.

00 66 =&gt; address of the bit read (66=read led test on the J2000RS/J3000)

Noted : 00 66 (decimal)=&gt; 42 (H)(hexadecimal)

01 =&gt; value of the bit read (here, 1) 1 to FF (H) possible is 255

00 =&gt; byte empty

44 46 =&gt; result of the calculation CRC16

**Writing of a word : Function 6**

Slave number	FUNCTION Number	ADDRESSES of word	Value of word		CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes		2 Bytes
01	06	00 14	00 03		168,8

Example :

01 =&gt; slave 1

06 =&gt; Number of function. Here the function to use is 6.

00 14 =&gt; address of the word to write (14=write the led state 14 on the J2000RS, not used on the J3000)

00 03 =&gt; value of the word writes (here, 3) 3 = 03 (H) is blinking

168 8 =&gt; result of the calculation CRC16

Reply :

Slave number	FUNCTION Number	ADDRESSES of word	Value of word		CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes		2 Bytes
01	06	00 14	00 03		168,8

Example :

01 =&gt; slave 1

06 =&gt; Number of function. Here the used function is 6.

00 14 =&gt; Address of the word written ( Noted : 2 bytes= 1 value )

00 03 =&gt; value of the word written (here, 3) 3 = 03 (H) led 14 of the panel 1 by blinking rapid

168 8 =&gt; result of the calculation CRC16

**Writing of N bits consecutive: Function 15**

Slave number	FUNCTION Number	ADDRESSES of first byte	Number of bit	Number of bytes	Value of bytes	CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes	1 Byte	N Bytes	2 Bytes

Reply :

Slave number	FUNCTION Number	ADDRESSES of first byte	Number of bits		CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes		2 Bytes

Example : See the list of example in the end of this document.

**Writing of N consecutive words: Function 16**

Slave number	FUNCTION Number	ADDRESSES of first word	Number of words	Number of bytes	Value of word	CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes	1 Byte	N Bytes	2 Bytes

Reply :

Slave number	FUNCTION Number	ADDRESSES of first word	Number of word	CRC16
1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes

Example : See the list of example in the end of this document.

**Error message:**

Slave number	FUNCTION Number received + 128	Error code	CRC16
1 Byte	1 Byte	1 Byte	2 Bytes

code of errors : 1= unknown function

2 = incorrect address (detection equally on the last addresses to process according to the Number of words or bits).

3= incorrect data

remark : an incorrect reception of CRC16, no message is returned.

**Number of slave " 0 " :**

All question including the number of slave " 0 " are taken by all slaves. No reply is returned.

**Maintenance :** Allow to know the slave number of a panel.

For the J2000 RS 485 : It can be used only if the machine is alone on the bus.

Demand : 65, CRC16 (2 bytes),

Reply : 65, slave number (1 byte), CRC16 (2 bytes),

**For the J3000RS485 :** It can be used only if the machine is alone on the bus.

To use the slave number 65 in the function demand " word Read " 3 or 4.

Demand : 65, 3 or 4, 130(2 bytes), 00 01(2 bytes),CRC16 (2 bytes),

Reply : 65, 3 or 4, 02,00 05(2 bytes), CRC16 (2 bytes),

**G) FRAMES EXAMPLES**

Following examples can be done on all A.M.I. products. Nevertheless, it is necessary to refer to the particularity of each one.  
 - some functions could not be installed in a model of product.  
 - products being different, addresses are not inevitably the same.  
 Example : It is possible to read the state of the led 18 on a J2100 RS 485, but not on the J3000 (including only 12 leds ).

**a) Reading of bits (Function 1 or 2) :**

Demand : N° slave, 1ou 2, address of 1st bit (2 bytes),NUMBER of bits, CRC16 (2 bytes),  
 Reply : N° slave, 1ou 2, 1, bytes containing one or the bits, CRC16 (2 bytes),  
 remarks : the reply is made always on 1 byte. 1st bit read is placed on the weak weight of the byte containing the bit. The bits unused are to 0.

Example :Reading of the state of the relay on panel 1 :

Demand : 01 - 01 - 00 64 - 00 01 - 252 30  
 Reply : 01 - 01 - 01 - 00 - 81 136 (The relay is not activated)

**b) Reading of words (Function 3 or 4) :**

Demand : N° slave, 3 or 4, address of 1st word, NUMBER of words (2 bytes), CRC16 (2 bytes),  
 Reply : N° slave, 3 or 4, NUMBER of bytes read(1 bytes), value of 1st word(2 bytes), ....., value of the last word (2 bytes),CRC16 (2 bytes),

Example : Reading of the led 8 state (Led by blinking rapid)

Demand : 01 - 03 - 00 08 - 00 01 - 5 200  
 Reply : 01 - 03 - 02 - 00 03 - 248 69

**c) Writing of a bit (Function 5) :**

Demand : N° slave, 5, address of the bit (2 bytes), value of the bit (1 byte), 0, CRC16 (2 bytes),  
 value of the bit=0, octet=00(H) is 0 value the bit=1, octet=FF(H) is 255  
 Reply : N° slave, 5, address of the bit (2 bytes), value of the bit (1 byte), 0, CRC16 (2 bytes),

1) with addresses of 01 (H) / 01 to 18 (H) / 24  
 Easy solution allowing to light the led in permanent : for the framework 0 = Off 1 = Lighted

Example : Led 4 Lighted :  
 FRAMEWORK of DEMAND  
 SLAVE NUMBER / 5 / Address = 00 04 / 255 / 00 / 205-251 (CRC)

FRAMEWORK of REPLY  
 SLAVE NUMBER / 5 / Address = 00 04 / 255 / 00 / 205-251 (CRC16)

Example Led 6 Off :  
 FRAMEWORK of DEMAND  
 SLAVE NUMBER / 5 / Address = 00 06 / 255 / 00 / 108-59 CRC16

FRAMEWORK of REPLY  
 SLAVE NUMBER / 5 / Address = 00 04 / 255 / 00 / 108-59 CRC16

2) with the address of 42 (H) / 66 :  
 Example : To make a led test on panel 1  
 Demand : 01 - 05 - 00 66 - 01 (255)- 00 - 44 46  
 Reply : 01 - 05 - 00 66 - 01 (255)- 00 - 44 46

3) with addresses of 64 (H) / 100 to C3 (H) / 195 :  
 Simple solution allowing to light the led of every available manners :

	Bit to 0	Bit to 1	example on led 1 (the value of the bit to 1 is 255)
Address +0 to +3=>	Off		100 bit 0 or 103 bit 0
Address +0 =>		Lighted	100 bit 1



Address +1	=>	Flash	101 bit 1
Address +2	=>	Rapid	102 bit 1
Address +3	=>	Slow	103 bit 1

**Addresses :**

LED 1 =Bits 100 to 103

LED 2 = Bits 104 to 107

LED 3 = Bits 108 to 111

LED 4 = Bits 112 to 115

.....

.....

LED 22 = Bits 184 to 187

LED 23 = Bits 188 to 191

LED 24 = Bits 192 to 195

**Example Led 4 Lighted Rapid :**

FRAMEWORK of DEMAND

SLAVE NUMBER / 5 / Address = 114 / 255 / 00 / 44-33 (CRC16)

FRAMEWORK of REPLY

SLAVE NUMBER / 5 / Address = 114 / 255 / 00 / 44-33 (CRC16)

**Example Led 22 Lighted slow :**

FRAMEWORK of DEMAND

SLAVE NUMBER / 5 / Address = 187 / 255 / 00 / 252-31 (CRC16)

FRAMEWORK of REPLY

SLAVE NUMBER / 5 / Address = 187 / 255 / 00 / 252-31 (CRC16)

**Example Led 1 Lighted flash :**

FRAMEWORK of DEMAND

SLAVE NUMBER / 5 / Address = 102 / 255 / 00 / 108-37 (CRC16)

FRAMEWORK of REPLY

SLAVE NUMBER / 5 / Address = 102 / 255 / 00 / 108-37 (CRC 16)

**Example Led 1 to extinguish after a flash :**

FRAMEWORK of DEMAND

SLAVE NUMBER / 5 / Address = 102 / 00 / 00 / 45-213 (CRC16)

FRAMEWORK of REPLY

SLAVE NUMBER / 5 / Address = 102 / 00 / 00 / 45-213 (CRC 16)

**ATTENTION :**

- To change the type of lighting on a led, it is not necessary to extinguish it. You can send directly the type of lighting wished. The led will display with the last type of lighting received.

- To extinguish a led, you can :

\* to return any address to the concerned led by indicating : value of bit =0.

**d) Writing of a word (Function 6) :**

Demand :Slave number , 6, address of the word(2 bytes), value of the word (2 bytes), CRC16 (2 bytes),

Reply : slave number, 6, address of the word (2 bytes), value of the word (2 bytes), CRC16 (2 bytes),

Example : To light every leds in flash on panel 1 :

Demand : 01 - 06 - 00 00 - 00 02 - 08 11  
 Reply : 01 - 06 - 00 00 - 00 02 - 08 11

Example : To light the led 14 of the panel 1 by blinking rapid :

Demand : 01 - 06 - 00 14 - 00 03 - 168 8  
 Reply : 01 - 06 - 00 14 - 00 03 - 168 8

**e) Rapid reading of 8 bits (Function 7) :**

Demand : slave number, 7, CRC16 (2 bytes),  
 Reply : slave number, 7, slave number , CRC16 (2 bytes),

Example : This function serves only ascontrol :

Demand : 01 - 07 - 65 226  
 Reply : 01 - 07 - 01 - 227 240

**f) Writing of N bit (Function 15) :**

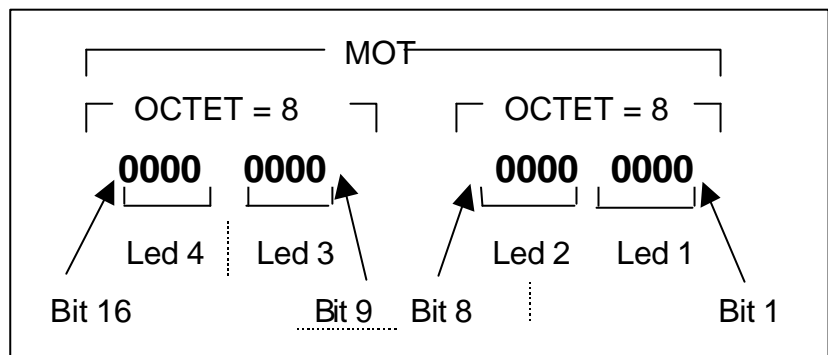
It is possible to program the totality of the leds in a single frame and start the framework to the address where you want, but each led is programmed on 4 bits, 2 leds by byte so 4 leds by 2 group bytes.

Demand : slave , 15, address of 1st bit (2 bytes), number of bits (2 bytes), number of bytes (1 byte), value of the bits (n bytes), CRC16 (2 bytes),  
 Reply : slave, 15, address of 1st bit (2 bytes), number of bits forced(2 bytes), CRC16 (2 bytes),

Value of the bit: 0 = Off 1 = Lighted 2 = Flash 4 = Rapid 8 = Slow

Value		on 4 BITS	
Off	=	0	is 0000
Lighted	=	1	is 0001
Flash	=	2	is 0010
Rapid	=	4	is 0100
Slow	=	8	is 1000

Address " bit " usable :  
 LED 1 = Bits 100 to 103  
 LED 2 = Bits 104 to 107  
 LED 3 = Bits 108 to 111  
 LED 4 = Bits 112 to 115  
 .....  
 .....  
 LED 22 = Bits 184 to 187  
 LED 23 = Bits 188 to 191  
 LED 24 = Bits 192 to 195



Example Led 1 blinking quick : 0100 is 04 written in decimal ( 4bits, 1 byte, values of the bits 04)

FRAMEWORK of DEMAND  
 SLAVE NUMBER / 15 / 00 100 / 00 04 / 01 / 04 / 78-157 (CRC16)

FRAMEWORK of REPLY  
 SLAVE NUMBER / 15 / 00 100 / 00 04 / 21-215 (CRC16)

Example Led 1 Lighted rapid + Led 2 Lighted Slow + Led 3 Lighted rapid + Led 4 Lighted Flash :

0010 0100 1000 0100 is 132-36 written in decimal ( 16bits, 2 bytes, values of the bits 132-36)

FRAMEWORK of DEMAND

SLAVE NUMBER / 00 100 / 00 16 / 02 / 132 36 / 137-31 (CRC16)

FRAMEWORK of REPLY

SLAVE NUMBER / 15 / 00 100 / 00 16 / 21-216 (CRC16)

**g) Writing of N words (Function 16) :**

Configuration in a framework of the totality of the leds and the user relay.

remark : The writing of configuration words is made always in the address 0019 (H) is 0-25(D) in decimal writing on 2 bytes and on 5 words.

Demand : slave, 16, 0025 (D) (2 bytes), 0005 (D) (2 bytes), 10 (D), value of 1st word, ....., value of the last word, CRC16 (2 bytes),

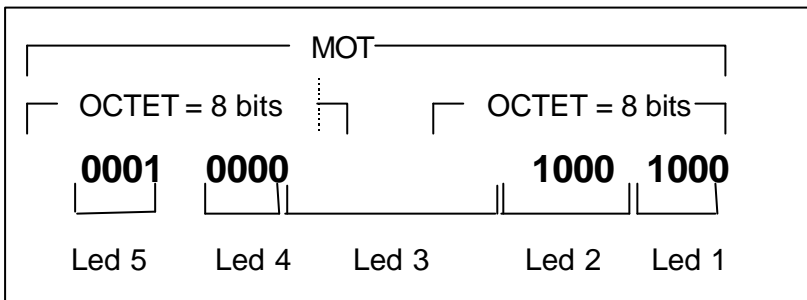
Reply : slave, 10 (D) / 16, 0025 (D) (2 bytes), 0005 (D) (2 bytes), CRC16 (2 bytes),

Example : Handwriting of 24 leds + the relay

Configuration wished :			in decimal binary	
Leds 1,4,6,10,20,22	= Off	=>	0	is 000
Leds 2,5,9,15,23	= Lighted	=>	1	=> 001
Leds 3,7,8,14,21	= flash Indicator	=>	2	=> 010
Leds 11,13,16,18	= rapid Indicator	=>	3	=> 011
Leds 12,17,19,24	= slow Indicator	=>	4	=> 100
Relay	= Asset	=>	1	=> 001

The configuration of each led is realized on 3 bits which allows to set 5 leds by words. The last bits of strong weight is not used.

1 Word = 2 bytes = 2x8bits is      xxx xxx xxx xxx



Each bit having for value :

32768-16384-8192 -4096	2048-1024-512-256	128-64-32-16	8-4-2-1
value of the word:	0 0 0 1 0 0 0 0		1 0 0 0 1 0 0 0
result :	4096	+ 128	+ 8 = 4232

4232(D) = 16-136 (D) written on 2 bytes = 10-88(H) in hexadecimal

To the final, the address of 5 firsts leds being 19(H) = 25(D), the value write in 2 bytes will be : 16 -136

Therefore for the address 19

Led 5 Led 4 Led 3 Led 2 Led 1  
 0 001 000 010 001 000 = 0001 0000 1000 1000 = 4232 = 16 136 (D)

For the address 20

Led 10 Led 9 Led 8 Led 7 Led 6  
 0 000 001 010 010 000 = 0000 0010 1001 0000 = 656 = 2 144 (D)

For the address 21

Led 15 Led 14 Led 13 Led 12 Led 11  
 0 001 010 011 100 011 = 0001 0100 1110 0011 = 5347 = 20 277 (D)

For the address 22

Led 20 Led 19 Led 18 Led 17 Led 16  
 0 000 100 011 100 011 = 0000 1000 1110 0011 = 2275 = 08 227 (D)

For the address 23

Relay Led 24 Led 23 Led 22 Led 21  
 0 001 100 001 000 010 = 0001 1000 0100 0010 = 6210 = 24 66 (D)  
 19(H) = 25 (D) is the address of the first word. The framework writes :

Demand : 01 - 16 - 00 25 - 00 05 - 10 - 16 136 - 2 144 - 20 227 - 8 227 - 24 66 - 176 7 (CRC)

Reply : 01 - 16 - 00 25 - 00 05 - 209 205 (CRC)

**h) Error messages :**

Number, function code received + 128, code of error, CRC16

1= unknown function

2 = incorrect address (detection equally on the last addresses to process in function of the NUMBER of words or bits).

3= incorrect datum

remark : on an incorrect reception of CRC16, no message is returned.

**i) Number of slave " 0 " :**

All ask them handwriting comprising the number of slave " 0 " are taken by all slaves. No reply is returned.

**J) Maintenance :**

Demand : 65, CRC16 (2 bytes),

Reply : 65, slave number(1 byte), CRC16 (2 bytes),

## **H) MS DOS HELP MEMORY**

<b>DIR</b>	Displayed on PC screen the contents of the current directory.
<b>DIR/p</b>	Displayed on PC screen the contents of the current directory, but page by page.
<b>DIR/W</b>	Displayed on PC screen the contents of the current directory, but in simplified version.
<b>CD..</b>	Go back up on the directory.
<b>CD xxx</b>	Go down under directory xxx. (think to put a space)
<b>Del *.*</b>	Erase all the contents in the directory xxx. (without space)
<b>MD XXXX</b>	Allow to create a new directory xxxx. (think to put a space)
<b>RD xxx</b>	Erase the directory xxx.
<b>a :</b>	Activate the reader <b>a</b>
<b>c :</b>	Activate the reader <b>c</b>

Attention in MS DOS :

-Think to configure your port :

To set your port RS232 (com1 or com2):

```
C:\AMI>MODE COM1 :9600,N,8,1 <ENTER>
```

You use the com 1 or 2 in 9600 bauds, without parity, 8 bits data, 1 bit stop.