Manuale di servizio

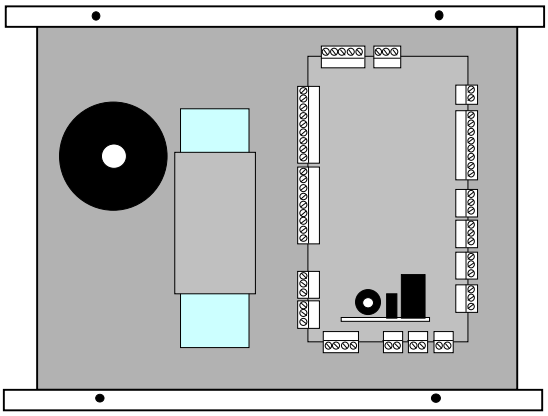
(Operating Manual)

Manual for

MULTI-FUNCTION CONTROL UNIT “COBRA1500-5000”

Ver. 01.03.01

Manual Rev. /a on 20.06.2002



N.B.: the contents of this manual are purely a **guideline** since the Producer reserves the right at any time to make any **modifications** it considers necessary to develop the product.

CONTENTS

1 Description of the Panel

Important note: in this manual we shall use the following terms indifferently to refer to the 'COBRA 1500-5000' board: "Panel", "Control unit", "peripheral" and "device".
The control unit is a peripheral device able to communicate with the Controller.
The Controller is a PC with Win95, Win98, WinNT, Win2000 or XP operating system on which the "WinGaep" Wincontrol program is installed or being executed.
It can be supplied in either a metal enclosure or in a plastic enclosure for outdoor applications.
One or **two card readers** of different technology can be connected to the control unit – i.e. magnetic, proximity, hands-free, optical (1 only) etc, with or without keyboard and display.
Every control unit has 6 inputs with various use, i.e. port status, sensors, photocells, alarm contacts, loops, etc, and 4 relay outputs utilised for electrical locks, activators, lighting, "traffic lights", etc.
The COM1 port must be used for connection to the management program.
This port is provided with two electrical interfaces:

- COM1 in RS485 (with galvanic separation)
- COM1 in RS232 (with galvanic separation)

Obviously using one excludes simultaneous use of the other.
Two protocols can be activated on the COM1 door:

- via Polling (**/P**) – default value
- via Modem/Terminal-Server (**/T**)

For connection to the "SpaceNet" network, use serial port COM1 - RS 485 with the Polling (/P) protocol.
For direct (point-to-point) connection between a PC and a Control unit, use the COM1 - RS 232 serial port with the Polling (/P) protocol.
To connect a PC to a remote Control unit, interface the latter with a Modem or with a Terminal-Server. On the Control unit, use the COM1 – RS 232 serial port, activating the protocol for the Modem/Terminal-Server (/T).
The information contained in this manual refers essentially to the standard operation for Polling (/P) using the "Space Net" network. The differences for via Modem (/T) operation will be explained in a dedicated paragraph.
The Control unit has an additional serial port. This can be used for updating the board's firmware.
In future, the additional serial port can be used for interfacing with a serial printer, optical reader, etc, according to case.
The Control unit is supplied with a "Clock Module" with a lithium buffer battery, enabling the control unit to combine local date/time to card reading - this function is particularly useful for Presence Detection and for activating time-band related relays without using the Controller.
As concerns the Presence Detection facility, the user should directly see the date/time of the "clock-in", and therefore, we advise you to use readers complete with keyboard /display.
The system normally operates with the Control units on-line - in this case, one refers to "On-line" mode.
This is the only mode ensuring all the functions possible with the program installed on the PC.
If communication with the Controller is interrupted, the Control unit switches to "Stand-Alone" mode.
Only some of the functions are possible in this mode, and, therefore, there are some limitations.

- in terms of the system
- in terms of the Control unit

A paragraph will describe these limitations.

1.1 Main features

Local memory for card archive, time-bands, holidays.
Archives sent from PC.
Lithium battery for maintaining data memory and clock/dater even in the event of a power cut.
Functional settings stored on a non-volatile memory (EEPROM).
Programming by PC of functional settings.
Local buffer memory for storing user codes and transactions: the partition (i.e. the space available for the user codes and transactions) is dynamic, that is, user-settable.
This means that the value of the maximum number of storable cards can be modified and, according to this, the Control unit calculates how much space remains for storing the events (Valid cards, alarms, etc) in Stand-alone mode.

There are **two** type of board according to the installed components:

1. **COBRA1500** (CPU with 2 serial ports; 128 KByte RAM)
2. **COBRA5000** (CPU with 3 serial ports; 512 KByte RAM)

A lithium battery is supplied to prevent loss of data in case of a power cut.
Facility for activating a reading halt if the transits memory is full.
RAM used partly for the transactions buffer and the remainder for archives (cards, time-bands, holidays).
Complete management of 1 or 2 separate gateways (reader A on one door and reader B on the other).
Maximum distance between a Control unit and a reader without keyboard / display: 100 mt.
Maximum distance between a Control unit and a reader with keyboard / display: 40 mt.
Power supply voltage: 12Vcc (-1V + 2V)
Maximum absorption: 900mA (varies according to number of connected readers).
Dimensions: 300x260x100mm if the metal enclosure is used.

1.2 “On-line” and “Stand-Alone”modes.

The Control unit is able to operate both by communication with the PC (“On-line” mode) and locally (“Stand-Alone” mode), if the communication with the PC is interrupted.
In other words, the Control unit has decision making ability and this makes it possible not to interrupt the Access Control or the Presence Detection service.
Switching between “On-line” and “Stand-alone” modes (and vice versa) is automatic depending on whether communication is activated with the Controller (polling) or not (if the communication mode is “/**P**”).
If the Control unit is in “/**T**” mode, because communication is via Modem or via LAN, it will be in “**Stand-alone**” mode only.
In “On-line” mode, every panel continuously dialogues with the PC (polling), transmitting every detected event in real-time. All decisions regarding a transit are delegated to the Controller (on which Wincontrol program is used) which records it in the “Historical Archive”. Before being enabled for transit, every card is subjected to all the controls specified in the “Access Control” (see relevant manual) program.
In “Stand Alone” mode, all decisions regarding transit by card, are made by the panel: all transits related data are recorded on the terminal’s non-volatile memory and are automatically downloaded toward the Controller, as soon as communication is restored.
Permission to open by the panel is based on personal-data archives also stored in the panel’s memory with lithium buffer battery.
The terminal’s memory capacity does not allow for complete management in “Stand-Alone” mode, as if the panel were “On-line” (communicating) with the Controller, but is sufficient to deal with a temporarily interrupted communication.

The panel can be programmed so that it is forced to operate always in "On-line" mode or always in "Stand-alone" mode (this will be described in detail later in this manual).
The always "On-line" mode is used for the following reasons:

- to emulate the behaviour of previous devices operating in “On-line” mode only
- to force the system to operate only if the PC is on-line in special applications.

If the Control unit is being polled by the PC, everything will operate as described; if not being polled, the card readings, alarms, etc., will be ignored, because local management is disabled.

The "Always Stand-alone" mode is in operation, even if communication with the Controller (active polling) is active.
In this case, if a card is read, the validity decision is made by Control unit – if it is valid, the transit data-item is sent to the internal buffer. If the Control unit is communicating with the Controller, the information is immediately transferred to the PC, or it will be as soon as communication is restored. Note the significant difference between the modes:

- if "On-line", the validity decision is made by the PC which immediately re-sends the outcome to the Control unit.
- if "Always Stand-alone", the decision is, in any event, made by the Control unit and is then communicated to the PC which does no more than archive it.

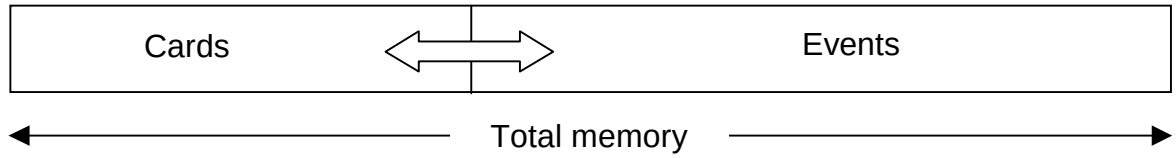
We shall now describe the functions of a panel in "Stand-alone" mode and its limitations compared to the "On-line" mode.

1.2.1 Characteristics in "Stand-alone" mode.

The following paragraph explains the memory limits of a Control unit in "Stand-alone" mode.

Historical archive (events) :

Set "**Max no. of cards**", and the Control unit calculates how many "**events**" can be accommodated in the remaining space; the calculation takes into account also if the "Justifications" are enabled or not.



The following tables summarise the possible partitions.
N.B.: with "**Justifications enabled**", the storable transactions are reduced by approximately 38% compared to the values valid with "**Justifications disabled**".

Cobra1500

Max No. of acceptable cards	No. of events ("Justif." = No)	No. of events ("Justif." = Yes)	Notes
10	12,038	7,524	Min cards – max events
500	10,372	6,483	Example
1,000	8,672	5,420	Example
1,500	**6,972**	**4,358**	**Preset situation**
2,000	5,272	3,295	Example
2,500	3,572	2,233	Example
3,000	1,872	1,170	Example
3,500	172	108	Max cards – min events

Cobra5000

Max No. of acceptable cards	No. of events (if "Justif," = No)	No. of events (if "Justif," = Yes)	Notes
10	51,360	32,100	Min cards – max events
1,500	46,294	28,934	Example
5,000	**34,394**	**21,496**	**Preset situation**
10,000	17,394	10,871	Example
15,000	394	246	Max cards – min events

N.B.: any card number value in the range can be set.

Memory storage is by a circular buffer; if it is full, the new events are overwritten over the old ones.

The Control unit can be set to halt readings if the buffer is **full** ("**Historical archive blocked**" function); in this case, to re-enable readings, download the data to the PC, thus emptying the buffer.
Data of **valid cards** only are normally archived in order to provide maximum space for this function: this means that the following are not archived: alarms, restore operations, invalid cards, etc.
However, you can enable memory storage of every **alarm** and every **restore** for inputs **ING1 – ING6 + Tamper**.
Storage of **valid cards** can never be **disabled.**

Other archives:

- Storing data of a maximum of **255 time-bands** (N.B.: numbered from 1 to 255)
- Storing data of a maximum of **32** days defined as "**Holidays**"

The Wincontrol program is not designed to send the "Access Level" archive to the Control unit and neither does it send – card by card - information on the associated "Access Level".
The Wincontrol program only sends the cards which are potentially valid at the Control unit.

This is typically effected according to the primary “Access Level” only. As from the 2.33.34 Version of the program, the other “Access Levels” associated with a card are also analysed - for further information, see the current program manual. However, the “Access Level” associated with the “Reason” is not used.
The Card archive is sent in incrementing numerical order according to the “Card Code”: first the lowest numbers and then the highest. As a result, only some of the cards in the archive (or even no cards at all) may be sent to a Control unit. The program sends all relevant cards to the control unit - it may send more cards than the specified capacity.
In that case, the Control unit stores the first cards received up to the maximum set number, ignoring any cards in excess of that number.

1.2.2 Operational functions in "Stand-alone" mode.

The following functions remain in operation in the "Stand-alone" mode (see the Wincontrol program manual for any explanations of the functions listed below):

- check of card validity per time-band and access level (typically primary level only)
- full use of the “character sequence” association
- validity of only the first “Installation Code” (the one on the left) of the three on the mask in the panel archive
- activation of the four outputs for valid cards (the valid card action message is not executed)
- different activation operations between reader A and reader B ("On-line" too in a local decision of the Control unit)
- activation of an output for invalid card (the invalid card action message is not executed)
- an Activation Card executes only the activation operations for that panel and not those for other panels (which, furthermore, are not on-line if "Stand-alone")
- PIN function in the two typing-in options: “type code + card” and “card + type code”
- PIN suspension in Time-Band + Always valid Exit
- PINSOST function, i.e. PIN replacing a card
- a “Reason” can be typed before presenting the card
- a “Justification” can be typed before presenting the card
- Activation of relay outputs according to time-band (OUT1 – OUT4 are activated if current time is within the specified time-band). N.B.: activation according to time-band is a decision made by the panel even in "On-line" mode
- Activation of one of the outputs for “Buzzer function”
- archiving of card data, date/time, reason and relevant reading head (A or B)
- activating operations associated with inputs providing they concern the same panel

1.2.3 Limitations and non-operational functions in "Stand-alone" mode.

In "Stand-alone" mode, the Control unit operates at a lower performance level – details below:

- limitation in recognising the set maximum number of cards
- limitation in storing the max. number of transits depending on max. selected number of cards
- card validity decision made locally according to the last archive download operation
- present/absent count not in real-time on Wincontrol program (however, the count is updated automatically while the historical archive is being downloaded as soon as the panel resumes "On-line" mode)
- “antipassback” function
- **action messages** of any kind as not downloaded in the Control unit memory
- subscription cards
- enabling / disabling cards for a certain period
- validity extended for subscription cards
- antipassback disabled for certain cards
- simplified access (rudimentary type)
- “Telepass-boost” + “Passive cards” combined management

1.3 Operating modes

The basic behaviour of the Control unit (the one that was set) is aimed at managing up to two readers of a **Pedestrian gateway** of an **Access Control** system not requiring any **further verifications** on the outcome of a transit.
We shall now explain in detail the concepts highlighted above:

A **Pedestrian** gateway differs from a **Vehicle** gateway equipped with vehicle presence magnetic loops, beams, etc.
Clearly, in the second case, we have a more specific application compared to a Pedestrian gateway, which is usually limited to closing a relay to activate the electric lock of a door; the opening of an electric gate also resembles more closely, in terms of setting, a Pedestrian gateway rather than a Vehicle gateway (in spite of the name).
The Control unit makes it possible to set an individual reader in **Parking mode** or otherwise. Therefore it follows that:

Reader A		ReaderB	
Park A = No (default)	**Park A = Yes**	**Park B = No (default)**	**Park B = Yes**
A Gateway: Pedestrian	**A Gateway: Vehicle**	**B Gateway: Pedestrian**	**B Gateway: Vehicle**

A paragraph will explain the Parking mode in detail.

If in On-line mode, it is the Wincontrol program that grants card validity; if in Stand-alone mode, the Control unit does this.
The card validation process in **Stand-alone** mode can be as follows:

- **"Access Control"** type with **complete** checks (existence of card in archive, Access Level, Time-Bands, etc.)
- **"Presence Detection"** type with **rudimentary** checks (Authorised Card code and Installation code)

A paragraph will describe the two modes in detail.

The reading of a valid card is usually sufficient to consider that the holder has transited through the gateway.
In some situations, it may be necessary to check if, after the card is read, the holder has actually entered/exited or has reversed. The Car Park manager or the management of a company may request the installed Access Control system to check if a vehicle or employee really entered/exited or not.
This can be done with the aid of external instruments such as turnstiles, magnetic loops, etc.,
Consequently, we must introduce the "**Transit**" concept, which completes the "**Card Reading**" concept.
"**Card reading**" is the event produced when the card is read.
"**Transit**" (if the function is enabled) is generated later.
In fact, a time-based mechanism is tripped whereby the Control unit waits for one or more events generating a "**Valid transit**" or an "**Invalid transit**". The events in questions can be closures of contacts, elapsing timeout, etc.
Detection of a "Transit" can be enabled per individual reader (A and/or B) and can be effected with either Pedestrian or Vehicle gateways: all that changes is the way of identifying Transit validity.
With "Transits" enabled, certain functions are updated no longer as "Valid card" but as "Valid transit".
E.g.: car park with local count and "transits" **disabled**: a valid card at entry immediately increases the counter of present and the "Present" status of the card itself.
E.g.: car park with local count and "transits" **enabled**: a valid card at entry causes beam opening only; if the card then correctly engages the Presence and Transit loops, a "Valid transit" is identified, which increases the counter of present and the "Present" status of the card itself. Otherwise, the count does not change.
Obviously, the Wincontrol program archives the "Transits" and manages the effects in the same way as the Control unit.
A paragraph will explain the "Transit" concept in detail.

1.3.1 "Access Control" mode

This is the standard operating mode. The card validity check is performed according to the Control unit's internal archives.
The last archives to be downloaded from the Controller apply.

- the maximum number of memory storable cards is equal to the **set value**
- the maximum number of events in the memory **corresponds to the max No. of cards**

In minimum security cases, the **Installation Code test can be disabled** (see paragraph).
If a card is not in the archive and has an "N" prefix, its data are, in any event, communicated to the Wincontrol program to enable the "Autoreading" function .

1.3.2 "Presence Detection" mode

If this mode is activated, and the Control unit is in "**Stand-alone**" mode or is forced into "**Always Stand-alone"** mode, it executes some very rudimentary tests when each card is read, to the extent that the Card archive need not be downloaded for it to function.
N.B.: - the **"T" prefix** must be active
- the **"Always Stand-alone"** function should be activated

Under these conditions, a card is considered valid if:

- the read **card number** is in the range from 1 to 65,535
- the read **installation code** is in the range 1 to 65,535 and coincides with that of Control unit

In practice, the card is checked to see if it belongs to the system and has a number within the specified range.
No tests are run on time-bands or for other purposes.
In minimum security cases, the **Installation Code test can be disabled**.

N.B.: by enabling this mode:

- the number of storable transits is automatically taken to the maximum by lowering the number of cards

- the prefix is forced to “T”

By re-setting the “Access Control” mode, the Control unit changes back the number of cards to the default value.
If communication to the Wincontrol program is re-established, the events data are automatically downloaded.

Important: the advantage of this mode is that the Control unit recognises **65,535** numbered cards from 1 to 65,535

1.3.3 “Transits”

In the introduction to this paragraph, we said that “Transit” is an information item generated when the Control unit is able to establish the outcome of an “OK Card reading”.
This is an actual event that is communicated to the Wincontrol program. If the Control unit is in Stand-alone mode, the event is archived and transmitted later when communication is re-established.
If the “Transit” function is enabled on a reader, “card reading” is not archived in Stand-alone mode but **only the “Transit”**. This is to avoid using too much memory space.
An “**OK Card reading**” may generate a “**Valid transit**” or an “**Invalid transit”**.
The communication of a “Transit” contains all the information contained in the “Card reading” (card number, installation code, reason, direction, date, time, reader, etc.).
As the transits identification process varies according to whether it is set as a “Pedestrian gateway” or a “Vehicle gateway”, the two cases will be explained in a specific paragraph.

1.3.3.1 “Transits” through a “Pedestrian” gateway

In a “Pedestrian” gateway (thus Park = No), a “**Valid transit**” is generated when an “OK Card reading” occurs and a certain entry is put into alarm status within a programmable time.
If this does not occur, when the time elapses, an “**Invalid transit**” is generated for that card.
The behaviour of the reader can be modified by changing the setting of “**Parking mode**” (N.B.: the same location as in the Parking mode is used, but there is no other connection).
This table summarises the implemented functions.

Parking Mode	Behaviour
1	Immediately communicates “Valid transit” after “OK Card reading”
2	Immediately communicates “Valid transit” after “OK Card reading”
3	Awaits entry enabling for “Valid transit”. If a second valid card is read during timeout, the first one immediately becomes an “Invalid transit” and the second one takes over from the first. The timeout is re-initialised.
4	Awaits entry enabling for “Valid transit”. Other cards are ignored during timeout.

Entries selected for validating “Transits”:

Reader A	Entry 1
Reader B	Entry 2

The entry can be felt on the leading edge and, therefore, a brief pulse of a voltage-free contact may suffice.
The typical application is to use the reader on a one-way turnstile. A person passing through causes the turnstile tripod to rotate which, in turn, closes a contact.

Possible settings for activating “Transits” for reader A.

Setting	Value
A transits management	Yes
Reader A car park	No
Reader A parking mode	From 1 to 4
A transits timeout	From 1 to 255

Possible settings for activating “Transits” for reader B.

Setting	Value
B transits management	Yes
Reader B car park	No
Reader B parking mode	From 1 to 4
B transits timeout	From 1 to 255

1.3.3.2 “Transits” through a “Vehicle” gateway

The connections of a vehicle gateway will be explained in the paragraph dedicated to the Car Park.
Here, we shall just state the following behaviour according to the mode being used.

Parking Mode	Behaviour
1	Immediately communicates “Valid transit” after “OK Card reading”
2	Immediately communicates “Valid transit” after “OK Card reading”
3	Waiting for Transit loop release for “Valid transit” If a second valid card is read during timeout, it is queued after the first one and waits to be managed.
4	For “Valid transit”, waiting for correct engagement sequence of Presence and Transit loops. If a second valid card is read during timeout, it is queued after the first one and waits to be managed.

Possible basic settings for activating “Transits” for reader A.

Setting	Value
A transits management	Yes
Reader A car park	Yes
Reader A parking mode	From 1 to 4
A transits timeout	From 1 to 255

Possible basic settings for activating “Transits” for reader B.

Setting	Value
B transits management	Yes
Reader B car park	Yes
Reader B parking mode	From 1 to 4
B transits timeout	From 1 to 255

Obviously, the Parking mode requires other settings as described in the relevant paragraph.

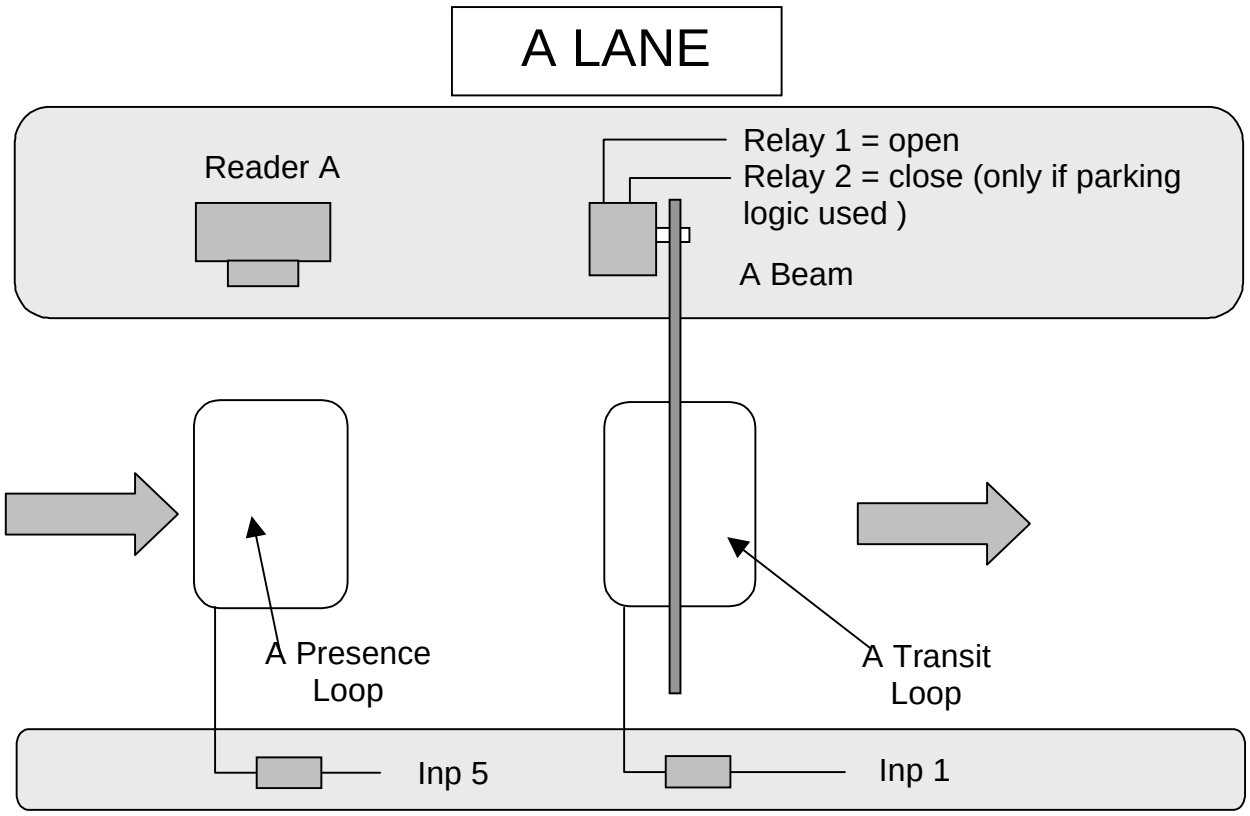
1.3.4 “Parking” Mode

This mode is used to manage a “**Vehicle gateway**” also known as a “**lane**”.
There are three types of possible lanes:

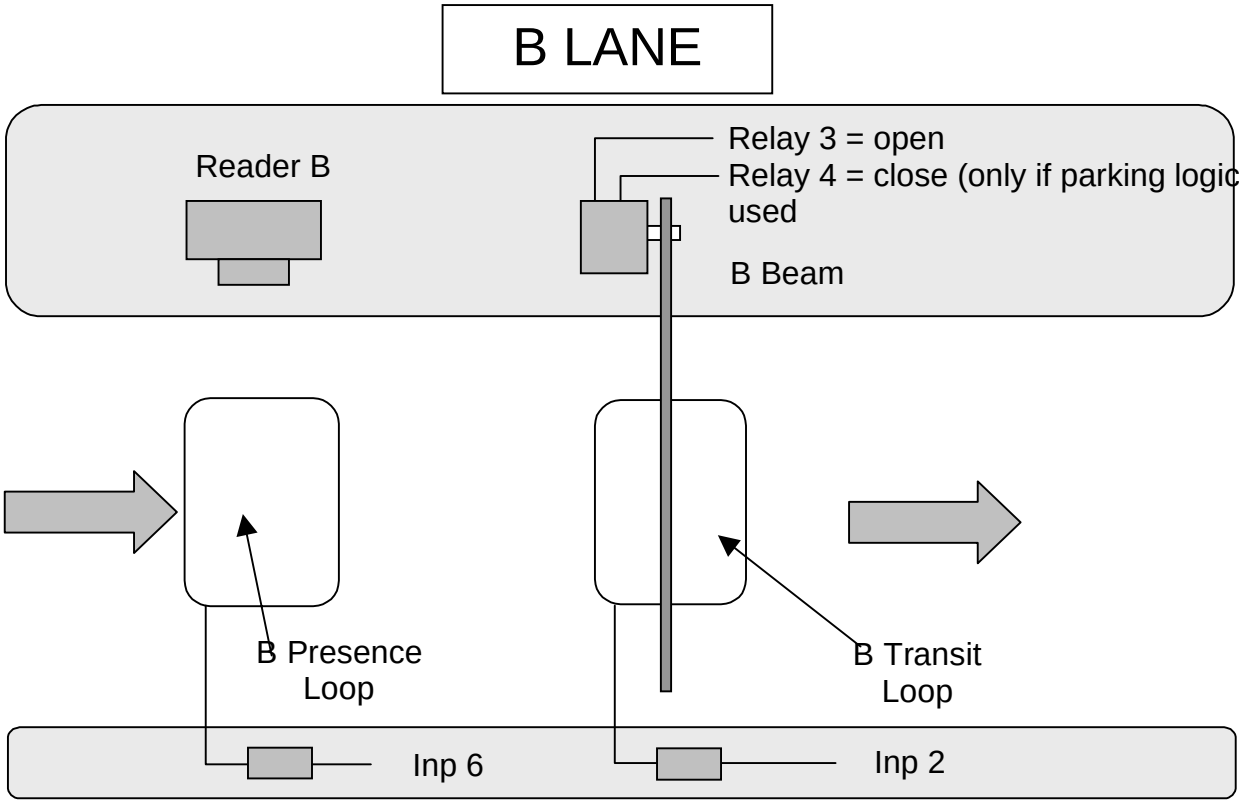
- “**A lane”** associated with reader A
- “**B lane**” associated with reader B
- “**single-beam**” lane, which is hybrid lane because it uses both readers and both travel directions (obviously authorised one at a time).

Remember that the following drawings show the lanes at their maximum expansion: some elements could be missing. In fact, no loops or just one loop are/is used in certain applications.

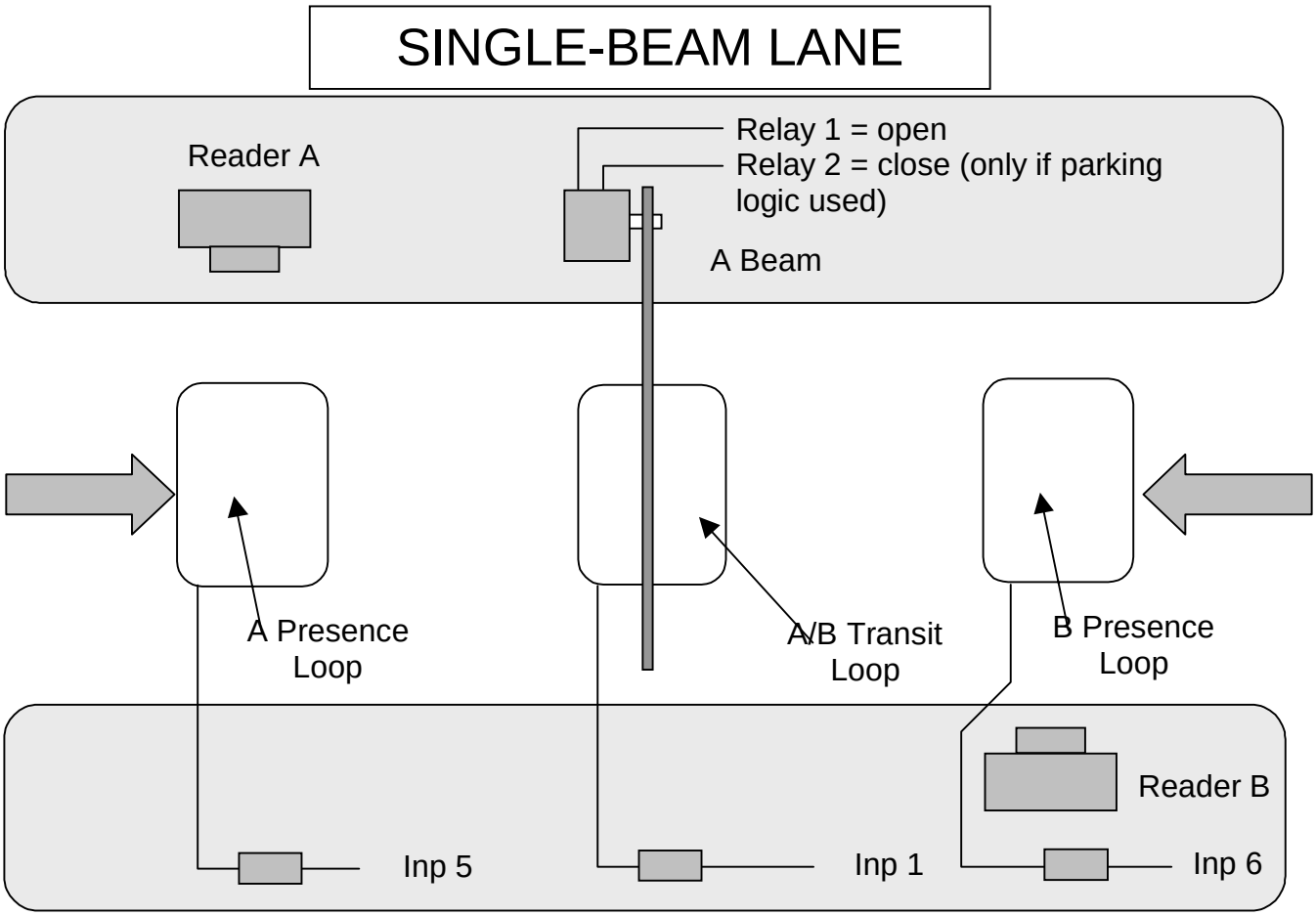
Lane A is shown in the following drawing.



B Lane is shown in the following drawing.



The “single-beam” lane is shown in the following drawing.



As can be seen from the drawings, certain **inputs** and certain **outputs** on the Control unit have a default use.
These two tables summarise the situation.

Inp 1	**A lane Transit Loop**
Inp 2	**B lane Transit Loop**
Inp 3	**No**
Inp 4	**No**
Inp 5	**A lane Presence Loop**
Inp 6	**B lane Presence Loop**

Out 1	**A lane beam opening**
Out 2	**B lane beam closing (*)**
Out 3	**B lane beam opening**
Out 4	**B lane beam closing (**)**

(*) only if A lane is in “Parking Logic”
(**) only if B lane is in “Parking Logic”

The following scheme illustrates a setting example in a situation where the Control unit has to manage two gateways, while piloting beams set in “**Parking Logic**”
Refer to the “Valid card activation ops.” item of the Panel archive mask in the Wincontrol program; use 4 brief times.

- OUT1 ➔ 4 tenths ☑☐ (if single-beam) OUT1 ➔ 4 tenths ☑☑
- OUT1 ➔ 4 tenths ☐☐ (if single-beam) OUT1 ➔ 4 tenths ☐☐
- OUT3 ➔ 4 tenths ☐☑ (if single-beam) OUT3 ➔ 4 tenths ☐☐
- OUT4 ➔ 4 tenths ☐☐ (if single-beam) OUT4 ➔ 4 tenths ☐☐

The following scheme illustrates a setting example in a situation where the Control unit has to manage two gateways, while piloting beams set in "**Normal Logic**"
Refer to the "Valid card activation ops." item of the Panel archive mask in the Wincontrol program; use 4 brief times.

▪	OUT1 ➔ 4 tenths	☑☐	(if single-beam)	OUT1 ➔ 4 tenths	☑☑
▪	OUT1 ➔ 4 tenths	☐☐	(if single-beam)	OUT1 ➔ 4 tenths	☐☐
▪	OUT3 ➔ 4 tenths	☐☑	(if single-beam)	OUT3 ➔ 4 tenths	☐☐
▪	OUT4 ➔ 4 tenths	☐☐	(if single-beam)	OUT4 ➔ 4 tenths	☐☐

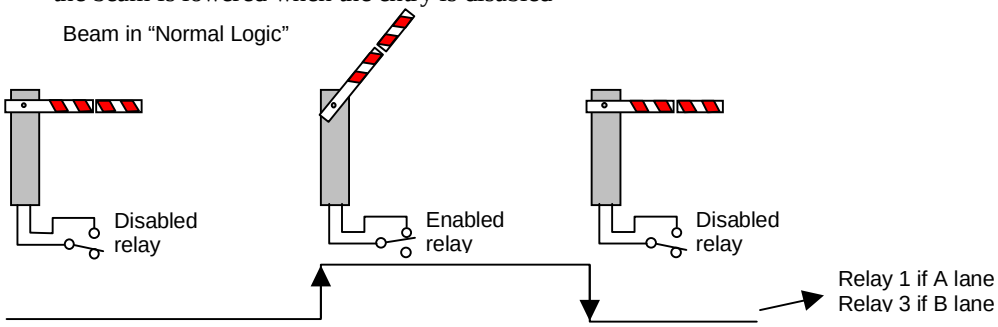
1.3.4.1 Beam logic

Beams with different types of activation can be correctly piloted.
Although there are various alternatives, the number of cases can be cut down to two situations:

1. beam in **normal logic**
2. beam in **Parking logic**

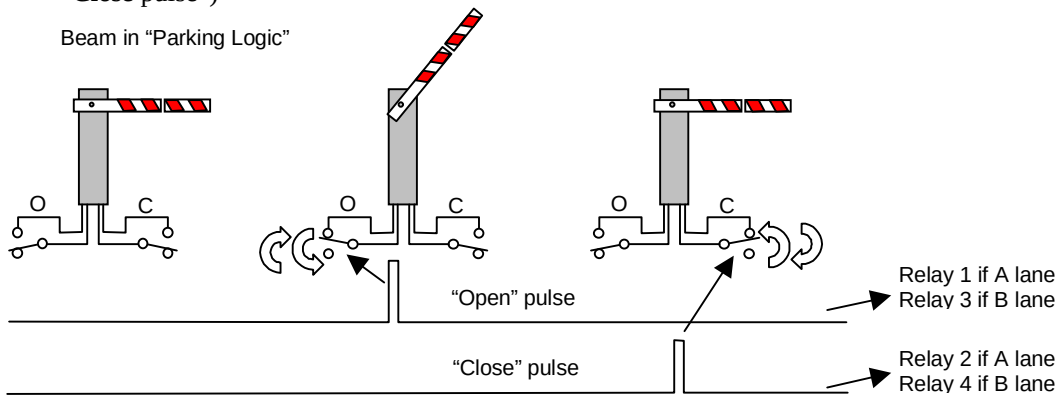
A beam in **normal logic**:

- has **only one** piloting entry
- the beam stays up until the input is activated
- the beam is lowered when the entry is disabled



A beam in **Parking logic**:

- has **two** piloting entries
- the entries are piloted by **pulses** (the relay must be enabled for a few tenths of a second)
- the beam is **raised** if there is a pulse on the opening entry (i.e. an "**Open** pulse")
- the beam is **lowered** if there is a pulse on the closing entry (i.e. an
- "**Close** pulse")



1.3.4.2 "Close" pulse every minute

If you are in "Parking Logic", you can enable the "**Close Pulse every minute**" function for each lane (A and B). It ensures that the beam is really closed when specified.
The pulse is activated per whole minute (sec. = 0).
Although it may be activated, it is not always performed. This is what typically happens:

- if in Timeout (by a valid reading or temporary opening)
- if Transit loop is engaged
- if in beam open in Time-Band (see below)

- if beam always open (see below)

1.3.4.3 Parking Mode

To take into account that, in certain applications, there are no loops, only one loop or both loops, we shall introduce the "**Parking Mode"** concept.
This mode may be set individually for each reader/lane: i.e. for "**A Parking Mode**" and for "**B Parking Mode**".
4 possible modes can be set:

Mode	Presence loop	Transit Loop
1	**No**	**No**
2	**Yes**	**No**
3	**No**	**Yes**
4	**Yes**	**Yes**

⇒ In mode 1) and 3), the card reader is usually enabled (as there is no vehicle Presence loop).
⇒ In mode 1) and 2), the beam is opened for a valid card and re-closed for Timeout (as there is no Transit loop).
⇒ In mode 2) and 4), the card reader is usually disabled and is enabled by the corresponding vehicle presence loop, and the result is:
- car absent → relay contact open (magnetic sensor) → entry disabled → unauthorised reading
- car present → relay contact closed (magnetic sensor) → entry enabled → authorised reading

⇒ In mode 3) and 4) the beam is re-closed under the following conditions:
- re-closed for Timeout if the Transit loop is disengaged and was never disengaged after the card was read
- re-closure before Timeout if the Transit loop was engaged and then disengaged
- re-closure after Timeout if the Transit loop was engaged at time expiry and then disengaged.

1.3.4.4 Presence loop and reader disabling
Enable **Parking Mode 2** or **4** to inform the Control unit that the "**vehicle Presence loop**" is connected.
This setting automatically forces the function "**Disable reader A**" or **B** in **Mode 1**.
See the **relevant paragraph** for explanations. The same mechanisms can be activated as "View text", etc.
May we remind you, at this point, that the disabling of a reader is linked to the status of an Input (Inp5 for reader A and Inp6 for reader B).
Obviously, in the Parking Mode, other functions are linked to the Presence (or Transit) loops, and not just reader disabling.

1.3.4.5 Loop Engagement/Disengagement times
One of the most important applications of the "Entry engagement time" and "Entry disengagement time" function is obtained in the Parking mode – these applications are described in a **specific paragraph**.
In fact, it may be convenient to set fairly long times for loop **disengagement** in order to ignore status variations caused by the movement of the vehicle on the loop.
Engagement times are set too, but are shorter.

1.3.4.6 A and B Transits timeout
The time assigned to each vehicle to transit through the gateway after producing a valid reading.
It can be set individually for each reader.
It must be correctly set, even if the "Transits" were not activated for a given lane, because it determines the beam re-closing time if the vehicle does not advance sufficiently to engage the Transit loop.
It is indicated in **seconds** and numbers in the range **from 1 to 255** are permissible.

1.3.4.7 Single-beam mode

This should be activated when a two-way lane with only one beam to open has to be implemented.
See the drawing illustrating the situation.
This is what you save in this configuration:

- space
- one beam
- one loop (the B Transit)

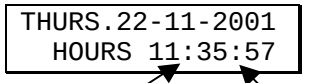
In this mode, the Control unit activates A beam only even in the case of valid cards on reader B.
Furthermore, it analyses the A Transit loop to check Transits tripped on side B of the lane.

1.3.4.8 A and B Transits management

We already mentioned Transit management on a Vehicle gateway (Car Park)
At this point, we should remind you that a "Valid transit" or an "Invalid transit" can be detected in a car park with a double loop.

1.3.4.9 Loop status display

If the view mode is **0 (zero)** (i.e. "always show **date/time**" if the presence loop is engaged or not), the loop engagement status can be seen on the display of the relevant reader (reader A display if A lane, reader B display if B lane).
Signalling is obtained by the separation character between hours and minutes (Presence loop) and between minutes and seconds (Transit loop): if the separator has **a colon** (":") = loop disengaged; if it has a semicolon (";") = loop engaged



"**:**" = **Presence** loop disengaged
";" = **Presence** loop engaged
"**:**" = **Transit** loop disengaged
";" = **Transit** loop engaged

1.3.4.10 Beam Mode

There are three possible modes for each beam:

1. **normal**
2. **always open**
3. **always closed**

The setting is typically commanded from the "**Push-button panel**" of the Wincontrol program.
However local modification is possible by using **keyboard programming** (if the keyboard/display is supplied).

The following are possible in the "**normal beam**" mode:

- normal vehicle transit by card reading
- the "**beam always open for T-B**" function (see below)
- the "**beam always closed for T-B**" function (see below)
- command from push-button panel: "**temporary opening**" (see below)
- command from push-button panel: "**temporary closing**" (see below)

The following are possible in the "**beam always open**" mode:

- beam always stays **open** until a new command is given
- **display** shows a text notifying unimpeded transit
- reader stays enabled (possibly linked to Presence loop status, if supplied)
- in "**normal logic**" the command relay is always active
- in "**parking logic**", the relay with the "**open pulse**" is active at the beginning and then every minute
- in "**parking logic**", the "**close pulse"** every minute is not however executed.
- normal vehicle transit with card reading is possible (even if this is unlikely with the beam already open)
- in "**parking logic**", the following command from the **push-button panel** is recognised: "**temporary opening**" (see below)
- it is given priority with respect to the "**beam always closed for T-B**" function (see below)

The following are possible in the "**beam always closed**" mode:

- beam always stays **closed** until a new command is given
- **display** shows a text notifying lane out of service
- the **reader** is, in any case, **disabled**
- in "**normal logic**" the command relay is always disabled
- in "**parking logic"**, the "**close pulse**" is enabled at the beginning and then every minute (even if not programmed).
- normal vehicle transits with card reading are not possible
- in "**parking logic**", the following command from the **push-button panel** is recognised: "**temporary closing**" (see below)
- it is given priority with respect to the "**beam always open for T-B**" function (see below)

1.3.4.11 Mode: "Beam open if in T-B"

Each beam can be made to stay up if in **Time-Band** (T-B).
This is possible only if in "**normal beam**" mode
Go to the "Activation ops. by time-band" item of the "Panel" Archive mask in the Wincontrol program.

- For **A beam**, indicate **Relay 1** + the required **Time-band**
- For **B beam**, indicate **Relay 3** + the required **Time-band**

If in the Time-band, all the functional considerations listed for the "**beam always open**" mode can be applied.
5 weekly activations/bands can be specified. This means that opening can be specified for several bands, but this limits the possibility of other settings.

1.3.4.12 Mode: "Beam closed if in T-B"

Each beam can be made to stay closed if in **Time-Band** (T-B).
This is possible only if in "**normal beam**" mode
Go to the "Activation ops. by time-band" item of the "Panel" Archive mask in the Wincontrol program.

- For **A beam**, indicate **Relay 2** + the required **Time-band**
- For **B beam**, indicate **Relay 4** + the required **Time-band**

If in the Time-band, all the functional considerations listed for the "**beam always closed**" mode can be applied.
5 weekly activations/bands can be specified. This means that closing can be specified for several bands, but this limits the possibility of other settings.

1.3.4.13 "Temporary" opening

The command is given from the "**Push-button panel**" of the Wincontrol program.
The result is:

- in "**normal logic**", the command relay is activated for the set timeout period
- in "**parking logic**", the command relay ("open") is activated for the pulse time
- the **Timeout** mechanism is recharged

The command is ignored if:

- in "**beam always closed"** mode
- in mode: "**normal beam**" – "**Beam closed if in T-B**"

1.3.4.14 "Temporary" closing

The command is given from the "**Push-button panel**" of the Wincontrol program.
The result is:

- in "**normal logic**", the command relay is activated for the set timeout period
- in "**parking logic**", the command relay ("open") is activated for the pulse time

The command is ignored if:

- if in **Timeout** (valid card reading pending or "temporary opening" command)
- in "**beam always open"** mode
- in mode: "**normal beam**" – "**Beam open if in T-B**"

1.3.4.15 Fraud events

"**Fraud events**" recognition by the Control unit can be activated individually on each lane.
The system is able to recognise two types of Fraud events.

Fraud event type	Associated code	Count influence
Queuing	**1**	**Yes**
Advance	**3**	**Yes**

Fraud event is an event which is immediately communicated if On-line or is stored and communicated as soon as the communication is restored.
Talking of Fraud events only makes sense if the reader is set for Parking and both the Presence and Transit loops are installed.
In fact, by analysing the loops engagement status, the system is able to understand the following:

- if a vehicle manages to **closely queue** to the preceding vehicle thus avoiding a card reading
- if a vehicle manages to transit by **advancing** along the lane without the card being read

Fraud events are typically anonymous in that the vehicle's card number cannot be identified as the card was not read. Furthermore, the vehicle may not even have a card.
An advance type fraud event can occur for various reasons, some of which are even lawful:

- the beam stays up because the mechanism failed
- the beam was knocked down
- the beam had received an "always open" command or "open if in T-B"

The result is, however, the same: a vehicle has passed through
If a "Count" is activated in the car park (or in the Control unit or in the Wincontrol program), this transit must be taken into account because it still influences the number of available places.
The "**direction**" each lane has in the case of a "Fraud event" can be set, i.e. if the lane is:

- **entry** (default for **A**)
- **exit** (default for **B**)
- **neutral** (no influence on count)

The "**Queuing fraud event**" or the "**Advance fraud event**" thus **increases** the count if they occur in an **entry** lane, and **reduce** it in an **exit** lane.
The Control unit's counting mechanism is similar to that of the Wincontrol program – this is how it works:
There are three counters:

1. **Total** (No. of occupied places in the car park)
2. **Present cards** (No. of cards in the archive that are "Present", i.e. correctly read cards).
3. **No. of Fraud events** (this number can be positive or negative; it refers to the number of irregular transits)

The total is the result of an algebraic sum (with a plus or minus sign) between lawful and illegal transits. That is:

"Total" = " Present cards" + "No. of Fraud events"

E.g.: The "Present cards" are 52; the fraud events are – 2 (i.e. there were more exit fraud events)
The "Total" is 50 occupied places.
The following are possible when programming from the keyboard:

- **reset** the "**Present cards**" number by setting **all** the cards as **absent** (N.B.: a data download can change the situation of each card)
- **align** the local count with respect to the car park situation

In fact, the program requires the number of vehicles actually counted inside the parking area, i.e. the "Total".
On the basis of this number, it calculates the "No. of fraud events" in order to balance the counts.
Before making the modification, it displays the result of these counts and asks for confirmation.

N.B.: the Wincontrol program must be checked to see if the fraud events we have so far described above are in the archive.
The fraud events must be inserted in the archive, referring to the "**Associated code**" specified in the table.
Next, you have to indicate which type of count they influence. For further details, see the relevant program manual.

1.3.4.16 Lane functionality
There are various types of transit.

Correct transit:

1. the vehicle engages the Presence loop (P)
2. the card is read and identified as valid.

3. the beam is raised
4. the Transit mechanism is tripped (if enabled)
5. Timeout is initialised
6. the vehicle goes forward and also engages the Transit loop (P + T)
7. as it moves forward, the vehicle disengages the Presence loop while remaining on the Transit loop (T)
8. finally, the vehicle also disengages the Transit loop
9. Transit is considered terminated and this is notified as a "Valid Transit" (if enabled)
10. The beam is re-closed when the Transit loop is disengaged

Note that:

- for as long as the vehicle is under the beam (Transit loop engaged), no beam closing commands are given (even if Timeout has elapsed).
- everything operates as above even if the vehicle transits before Timeout elapses (Timeout is reset when the Transit loop is disengaged)

Correct transit with two closely queued vehicles:

In this situation, a second vehicle is validly read while the first vehicle is still on the Transit loop (point 7). The mechanism is the same as up to point 9 in "Correct transit". It then becomes:

- the beam stays open
- Timeout is re-initialised
- the data of the second vehicle take over from those of the previous vehicle

From this moment on, the situations are the same as in the case of the first vehicle.

Invalid Transit due to Timeout:

1. the vehicle engages the Presence loop (P)
2. the card is read and identified as valid
3. the beam is raised
4. the Transit mechanism is tripped (if enabled)
5. Timeout is initialised
6. vehicle does not move until Timeout elapses
7. Transit is considered terminated and this is notified as an "Invalid Transit" (if enabled)
8. the beam is re-closed when Timeout elapses

Invalid Transit due to reversing:

1. the vehicle engages the Presence loop (P)
2. the card is read and identified as valid.
3. the beam is raised
4. the Transit mechanism is tripped (if enabled)
5. Timeout is initialised
6. the vehicle reverses and disengages the Presence loop
7. Transit is considered terminated and this is notified as a "Invalid Transit" (if enabled)
8. The beam is re-closed when the Presence loop is disengaged

Advance fraud event:

1. the vehicle engages the Presence loop (P)
2. the vehicle goes forward and also engages the Transit loop (P + T)
3. as it moves forward, the vehicle disengages the Presence loop while remaining on the Transit loop (T)
4. finally, the vehicle also disengages the Transit loop
5. the process identifying the "Advance fraud event" (if enabled) is considered terminated.
6. the beam is re-closed after the Transit loop is disengaged (unless it is "always open")

There are no card readings but only loop engagement.

Close queuing is acceptable in this case: the first vehicle reaches phase 3 (T), and a second vehicle can begin the process by engaging the Presence loop. When the first vehicle concludes transit by abandoning the Transit loop, the second vehicle takes its place in terms of management.

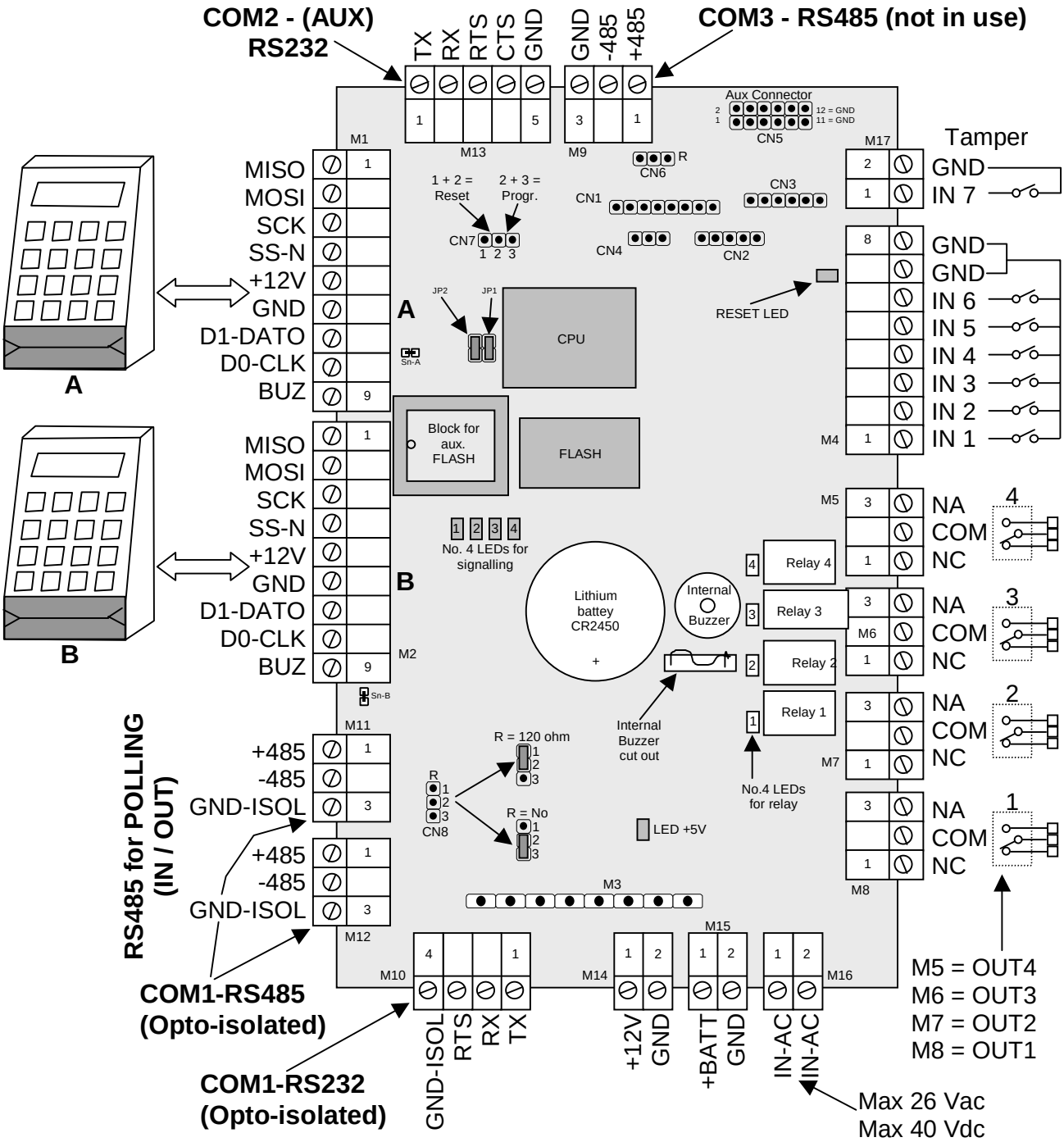
Close queuing fraud event:

The descriptions up to point 7 (T) in "Correct transit" apply. This is followed by:

1. the Presence loop (P) is engaged
2. no card reading takes place
3. the Presence loop is once again disengaged
4. the process identifying the “Close queuing fraud event” (if enabled) is considered terminated.

Note that:
the process described here could also refer to a second vehicle which engages the Presence loop and then reverses thus disengaging it. With only two loops, the two cases cannot be distinguished, but it is preferred to treat the event as a “Close queuing fraud event” as this is statistically more probable.

2 Connections and lay-out



The **COM1** serial port has two electric interfaces: **RS485** and **RS232**. Both have "**galvanic separation**" (Opto-isolated). Therefore the GND-ISOL of terminals M10, M11 e M12 is **not** connected to other GNDs.

LED + 5V must be normally **lighted**.
The **RESET LED** must be normally **OFF** (lighted at power-up only or during a reset).

The Control unit consists of a board which contains all the electronics and the connection terminals.
There are several functional logic blocks on the Control unit. They typically communicate with the outside world via terminals. The blocks are:

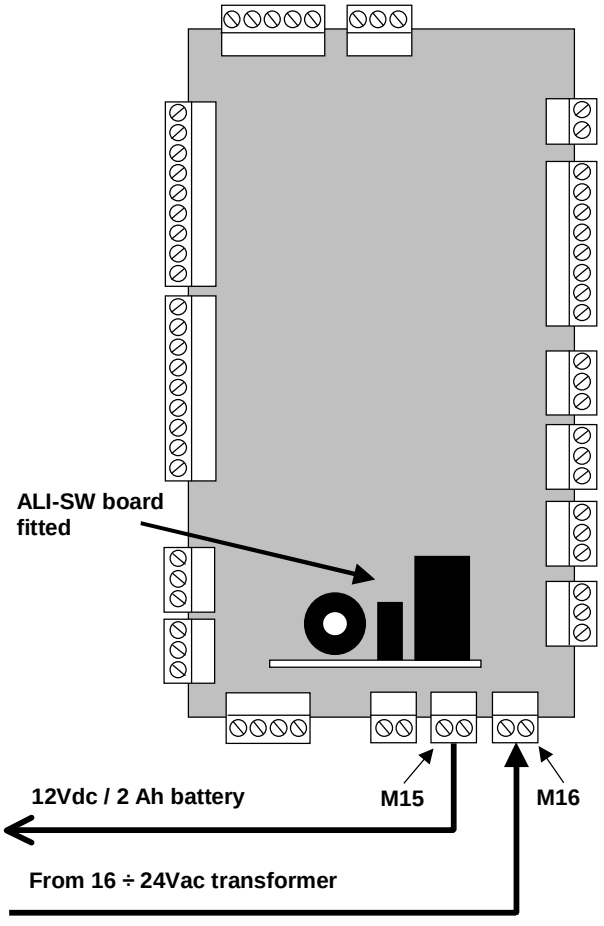
- power supply unit
- serial port for "polling" in RS485 for SpaceNet (COM1-485)
- Serial port for "polling" in RS232 (COM1-232)
- Auxiliary RS232 (COM2) serial port.
- Auxiliary RS485 (COM3) serial port, currently not managed
- Connection for reader A
- Connection for reader B
- physical inputs
- outputs

Power supply
The associated terminal boards are:

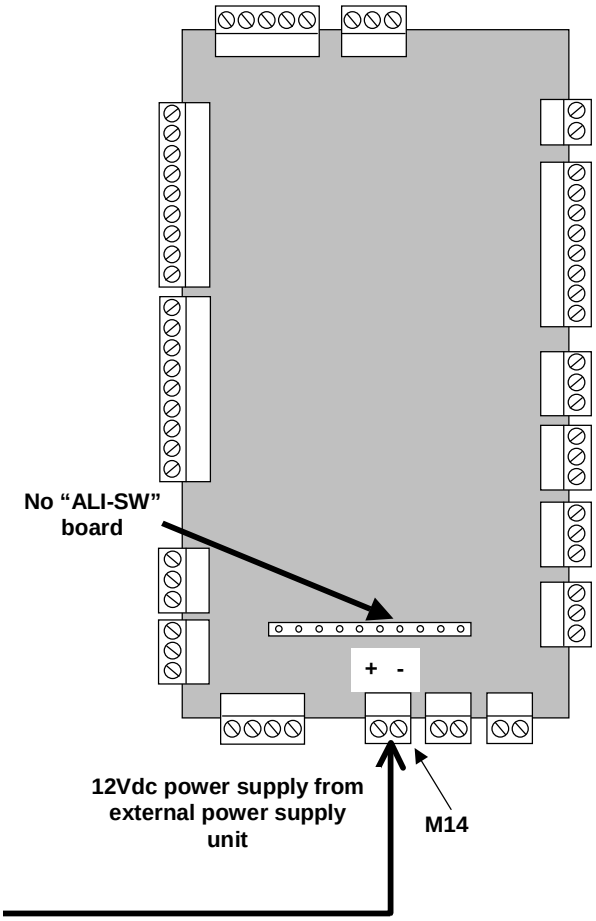
- M14: for direct input at 12Vdc (11.3 – 14 Vdc)
- M15: for connecting a 12V/2Ah battery (only if the "ALI-SW" board is fitted)
- M16: for connecting a 16 – 24 Vac transformer (only if the "ALI-SW" board is fitted)

Two alternatives are possible depending on whether the Control unit includes the "ALI-SW" board.
The following diagrams illustrate the two possible connection types.

Example of power supply using a transformer
N.B.: only if "ALI-SW" board is fitted



Example of 12 Vdc power supply using an external direct current power supply unit

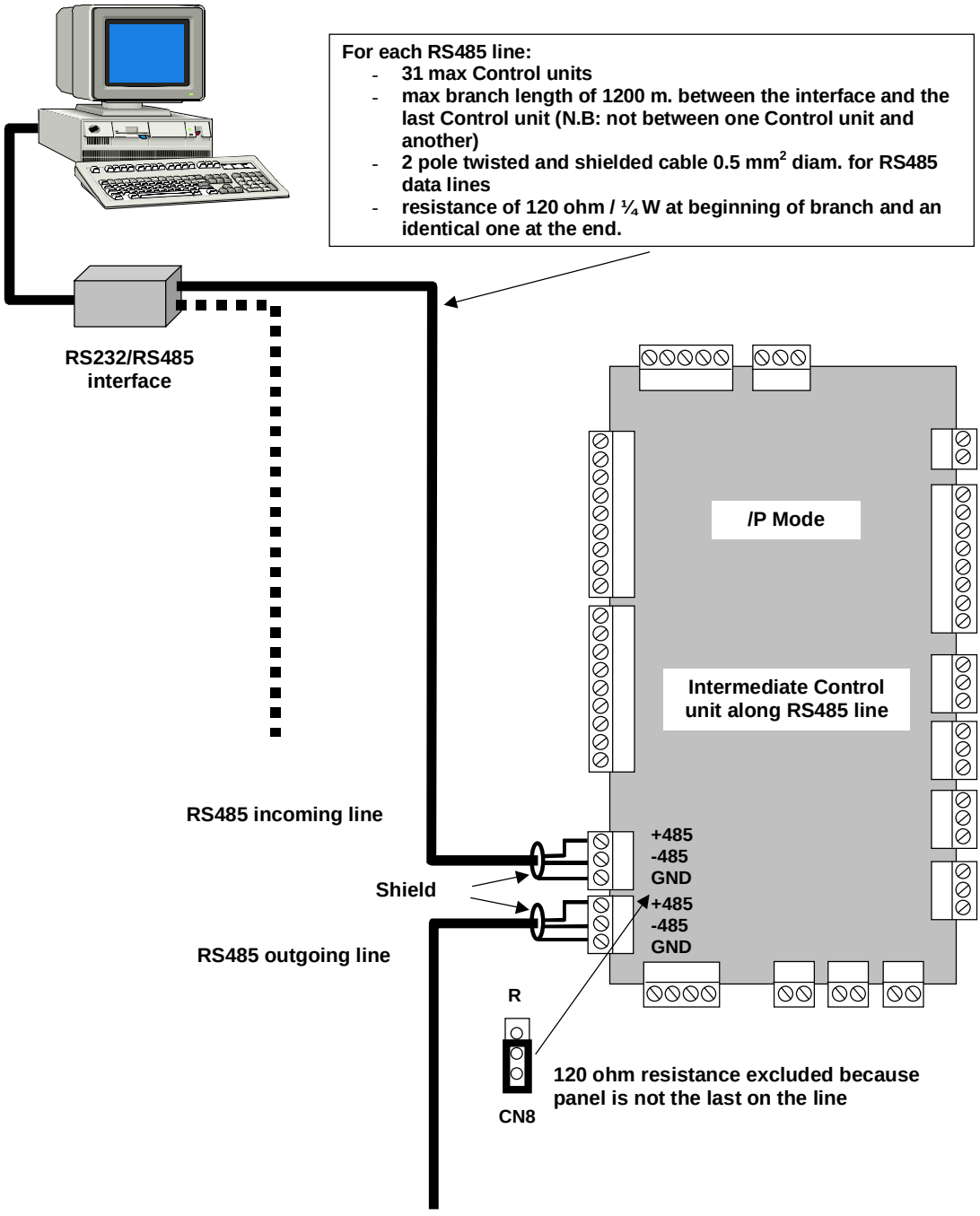


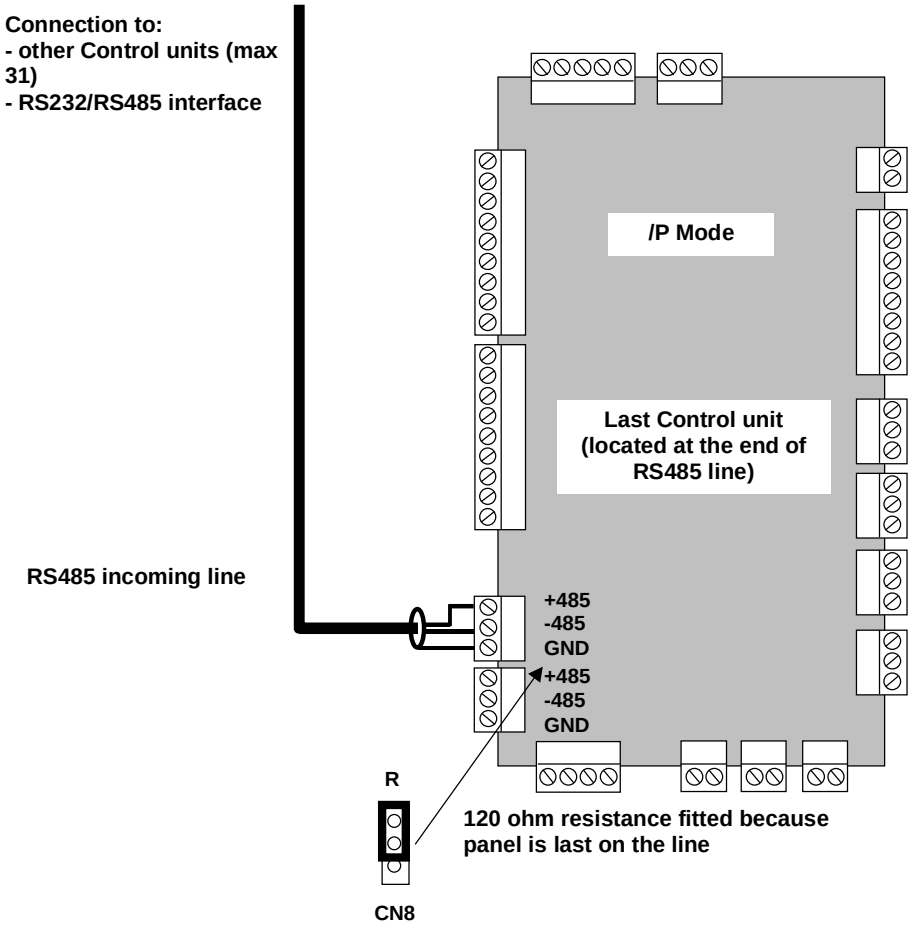
Serial port for "polling" in RS485 (SpaceNet)
The associated terminal boards are M11 and M12.
This port can be used only if the COM1-232 port is not being used.
Connect as described in the figure of the Control unit.
The serial port is opto-isolated. Two 3-pole terminal boards are provided – they are identical to each other because the corresponding Pins are connected in parallel: use one terminal board for the incoming line and the other for the outgoing line.
Meaning of Pins:

- Pin1: +485
- Pin2: -485
- Pin3: Cable shield

N.B.: the shield is isolated and is not connected to the board's GND: it serves only for anchoring the braids of the data-line cables.

Example of connection using "SpaceNet" in RS485

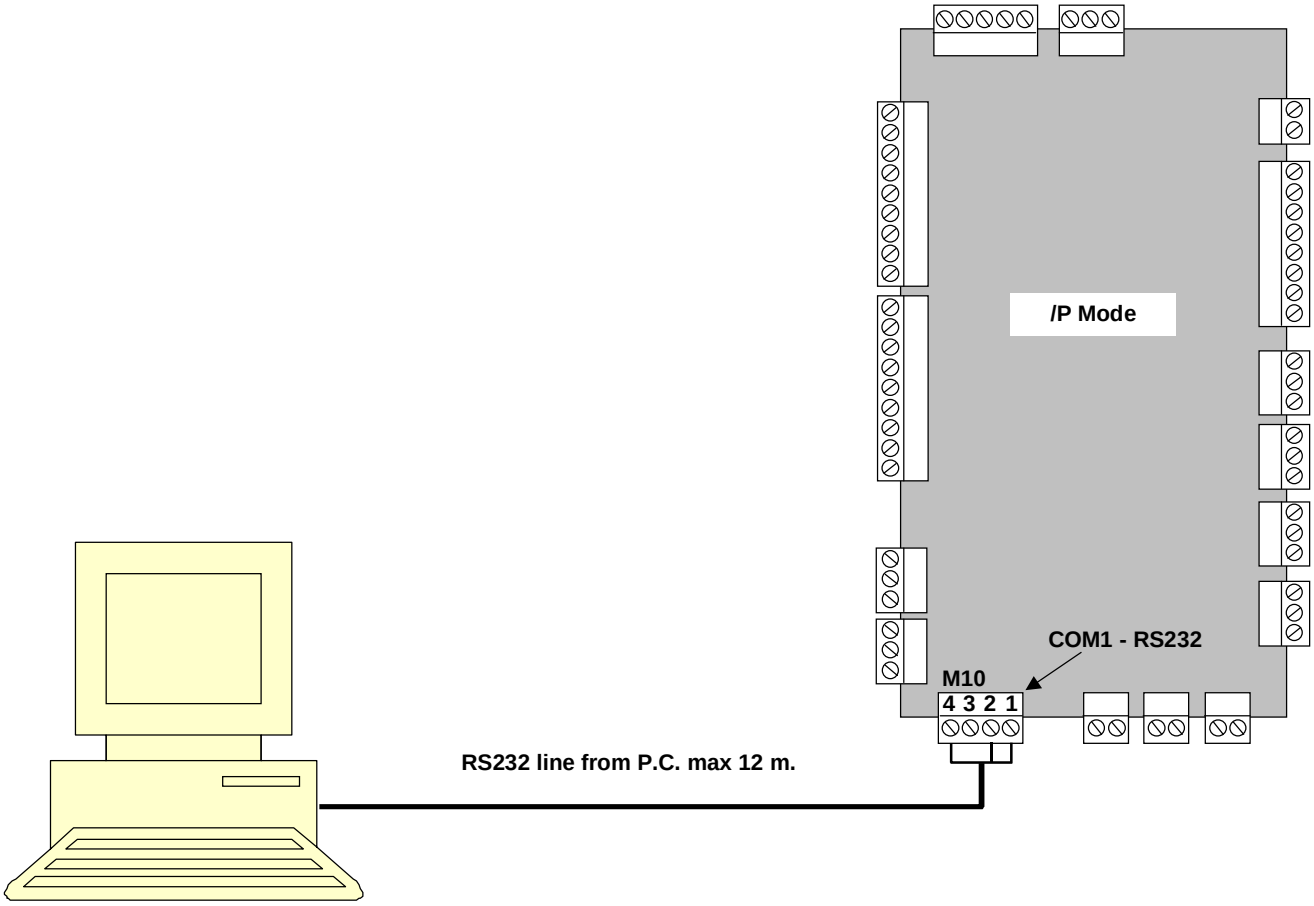



Connection to:
- other Control units (max 31)
- RS232/RS485 interface
/P Mode
Last Control unit (located at the end of RS485 line)
RS485 incoming line
+485
-485
GND
+485
-485
GND
R
120 ohm resistance fitted because panel is last on the line
CN8

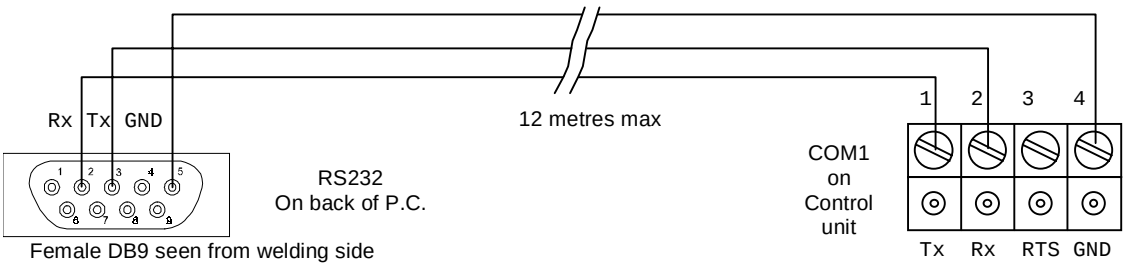
Serial port for “polling” or “modem” in RS232 (COM1-232)
The associated terminal board is M10.
This port can be used only if the COM1-485 port is not being used.
Connect as described in the figure of the Control unit.
The serial port is opto-isolated.
Meaning of Pins:

- Pin1: Tx
- Pin2: Rx
- Pin3: RTS
- Pin4: GND (isolated from the other GNDs on the board)

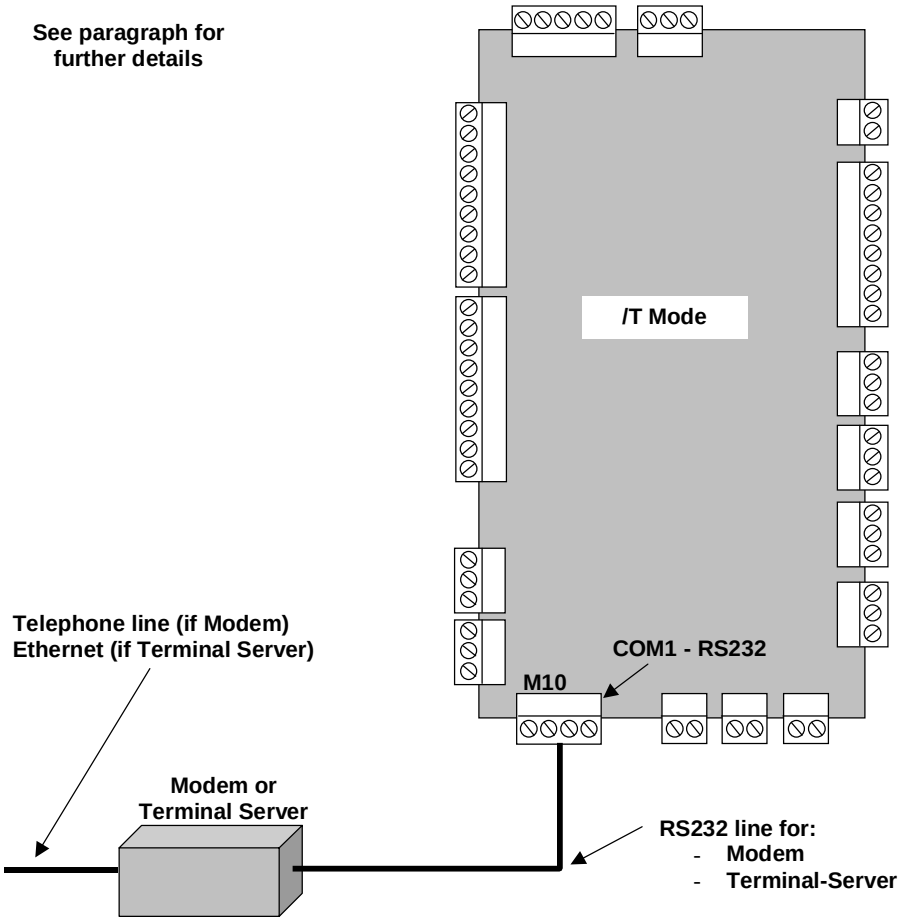
Example of direct connection to PC in RS232



Lay-out of cable to be made for the previous connection



Example of a connection to a “Modem” or to a “Terminal-Server in RS 232



The connection is described in detail and commented in the paragraph dedicated to “via Modem” or with “Terminal-Server” operation.

Auxiliary RS232 (COM2) serial port.
The associated terminal board is M13.
This port is used.

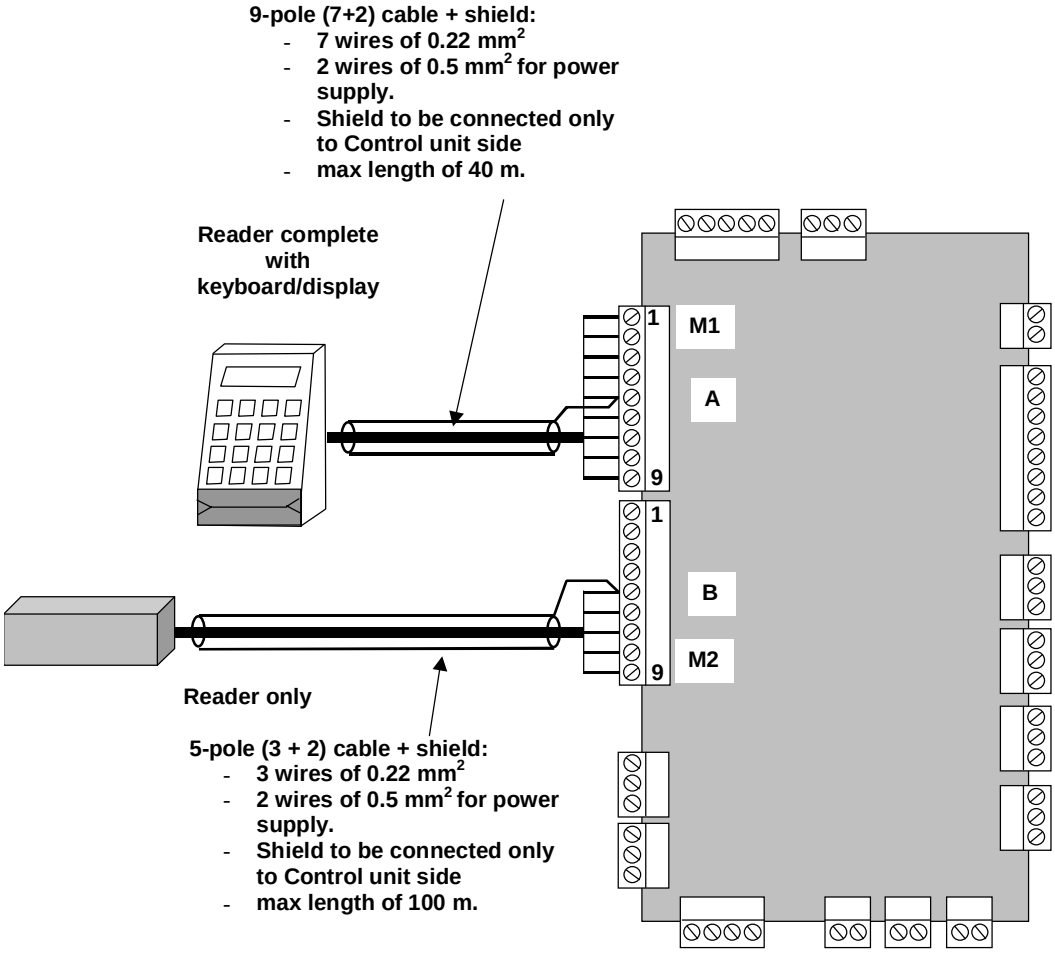
- for on-line printing of OK cards
- for updating of firmware (see specific paragraph).

This terminal board can be used in future to enable communication with peripheral devices such as optical readers, printers, etc.

Auxiliary RS485 (COM3) serial port.
The associated terminal board is M9.
This port is **not used** for the time being.
It is not implemented on the simpler version of the COM3 Control unit (the terminals are provided but are intended for other uses)

Connection for reader A
The associated terminal board is M1.
The Control unit can be programmed to read either the Wiegand codes or the Magnetic Stripe from B.
Note that Wiegand is the output standard for Cotag readers, whereas Magnetic Stripe is the standard for Credit Card type coded cards (ABA) or passive transponder readers.
The keyboard/display is built into certain types of readers (see catalogue).
The maximum distance for this type of connection is 40 metres if the keyboard/display is provided, and about 100 metres for card readers only.

Connection for reader B
The associated terminal board is M2.
The Control unit can be programmed to read either the Wiegand codes or the Magnetic Stripe from A.
Note that Wiegand is the output standard for Cotag readers, whereas Magnetic Stripe is the standard for Credit Card type coded cards (ABA) or passive transponder readers.
The keyboard/display is built into certain types of readers (see catalogue).
The maximum distance for this type of connection is 40 metres if the keyboard/display is provided, and about 100 metres for card readers only.



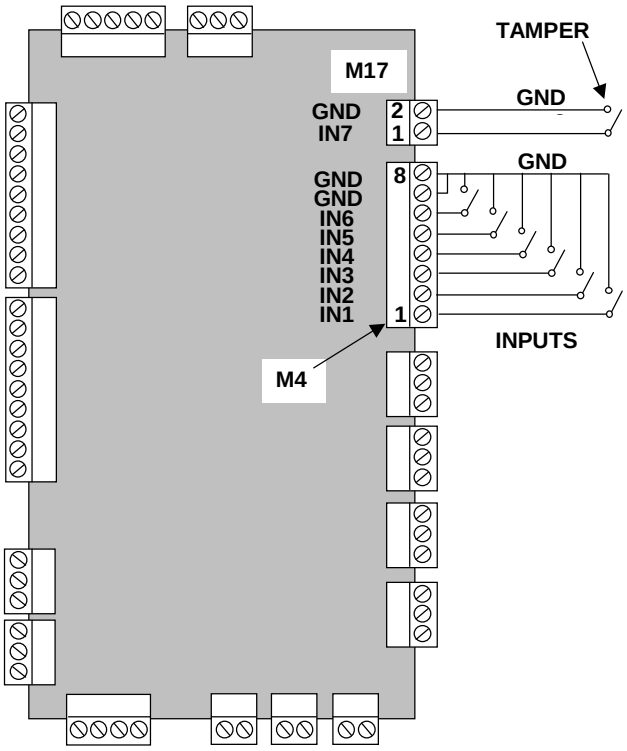
Inputs:
The associated terminal boards are M4 and M17.
M4 is an 8-pole terminal: it is used for inputs from 1 to 6.
M17 is an 2-pole terminal: it is used for input 7, usually dedicated to the Tamper.
Total inputs are 7. They should be activated by connecting them to GDN and using a voltage-free contact.
The **function logic** can be programmed, i.e. to be activated by “normally open” or “normally closed” contacts. The following table describes the default settings and the meanings of Pins.

TERMINAL	INPUT	STATUS WITH DEFAULT LOGIC AND FREE TERMINAL	**DEFAULT LOGIC**
M4-Pin1	IN1	Restore	☐ Norm. open
M4-Pin2	IN2	Restore	☐ Norm. open
M4-Pin3	IN3	Restore	☐ Norm. open
M4-Pin4	IN4	Restore	☐ Norm. open
M4-Pin5	IN5	Restore	☐ Norm. open
M4-Pin6	IN6	Restore	☐ Norm. open
M4-Pin7	GND		
M4-Pin8	GND		
M17-Pin1	IN7	Alarm	☑ Norm. closed
M17-Pin2	GND		

- A "**Normally open**" logic means that:
  - there is no tick-mark ☐ on the Wincontrol program's "Initialising On-line panels" screen mask
  - the input left disconnected is in Restore status.
  - the input connected to GND is in Alarm status.

- A "**Normally closed**" logic means that:
  - there is a tick-mark ☑ on the Wincontrol program's "Initialising On-line panels" screen mask
  - the input connected to GND is in Restore status.
  - the input left disconnected is in Alarm status.

In default logic, every M4 / M 17 input left disconnected is in restore status; to put it into alarm status, it must be connected to GND.
The tamper input is normally in alarm status: to put it in restore status, it must be connected to GND
Some inputs can change to an alternative function according to setting or EEPROM version.

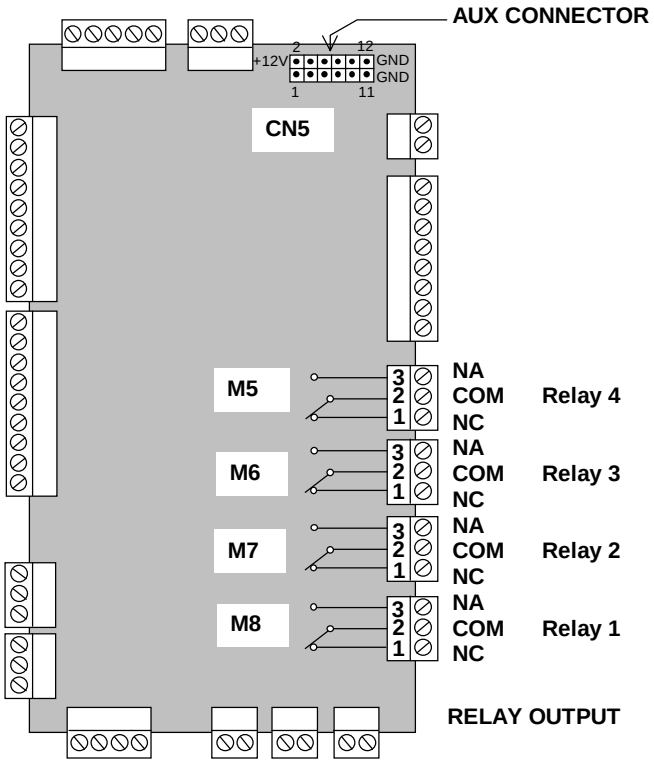


Outputs
The associated terminal boards are M8, M7, M6 and M5.
4 outputs are available. They have no default function and, therefore, can be utilised at the user's discretion. A valid card can activate all 4 outputs or part of them at a time, settable for each. Everyone of the 4 control outputs can be activated, in time-band too. If a valid card activates an already enabled output because the latter is within a time-band, when the relevant time elapses, the output remains activated because the timed activation command is ignored and the time-band is given priority.

- OUT2 activates RELAY 1; available on voltage-free terminal 1
- OUT2 activates RELAY 2; available on voltage-free terminal 1
- OUT2 activates RELAY 3; available on voltage-free terminal 1
- OUT4 activates RELAY 4; available on voltage-free terminal 1

N.B.: If connecting one ore more outputs of the SA 64 panel using direct current electro-operators, make sure that these devices are equipped with adequate **protective diodes** in parallel, and have a different power supply from that of the panel itself.
Noise induced by external operators could reset the Control unit's CPU at every activating/de-activating operation.
Remember that different activating operations can be defined for readers A and B: a paragraph will describe how to proceed.

N.B.: a **LED** is associated with each relay, which is lighted when the corresponding relay is active.



Auxiliary Connector: description of Pins

CN5

CMOS level (0- 5V)	OUT5-AUX	1	2	+12V
CMOS level (0- 5V)	OUT4-AUX	3	4	IN4-AUX
CMOS level (0- 5V)	OUT3-AUX	5	6	IN3-AUX
CMOS level (0- 5V)	OUT2-AUX	7	8	IN2-AUX
Open-Collector	OUT1-AUX (*)	9	10	IN1-AUX
	GND	11	12	GND

(*): OUT1-AUX is an Open-Collector type output.

Outputs from OUT2-AUX to OUT5-AUX are at level CMOS (0- 5V).

Inputs from IN1-AUX to IN4-AUX should be activated with voltage-free contacts toward GND.
Pins 11 and 12 are nearest to the hole for securing the board in the corner.

3 Card reading

One of Control unit's most important functions is card reading.
The control unit has no built-in readers but has inputs for interfacing with a maximum of 2 readers: reader A and reader B.
Each input consists of 2 PINs named D0-CLOCK and D1-DATO.
The program is able to decode two types of readers:

- readers with Magnetic-Stripe output (default setting)
- readers with Wiegand output

Each reader can be individually set on the Control unit. Example: reader **A** as **Magnetic Stripe** and reader **B** as **Wiegand**.

3.1 Type of card or transponder

Before explaining the different decoding methods, we prefer to briefly illustrate the various types of card or transponder (and the relevant readers) which can be connected to the Control unit, and summarise the main configurations to set in order to help those not familiar with these systems.
Remember that the following settings are not the only ones possible - they are just the most important ones.

3.1.1 Magnetic cards

Cards with a **univoca**l "Installation code" on the **magnetic stripe** and also a **sequential** or known "Card code"
The following basic settings are recommended

- Type of code decoding: **M** (Magnetic stripe)
- Prefix: **T**

Cards may be loaded in the PC's archive only if their data are known (Card Code and Installation Code).
If the reader is of the **Insertion** type, we advise you to use "Ignore card" for at least 5 seconds.

If the cards already existed and the programming criteria are unknown, act as in the case of the "Passive proximity cards". If the reader is of the **60% Insertion** type remember the following:

- you are advised to cut it down to 18 characters (Character sequence end)
- disable the **LRC** test (only if necessary)

3.1.2 Passive proximity cards

The following basic settings are recommended

- Type of code decoding: **M** (Magnetic stripe)
- Prefix: **N**

Autoreading is necessary to **load** the cards in the PC's archive,
If the reader is of the "double passive" type, also set the various types of reading (NA or RA – NB or RB) to distinguish Reason, Direction or both according to the desired effects.

3.1.3 Passive proximity Key-case transponders

Exactly the same rules as described for "Passive proximity cards" apply here.

3.1.4 Long-distance transponders for vehicle gateways

These are the Gigahertz transponders for long distance reading (a few metres) mostly for vehicle applications.
The following basic settings are recommended

- Type of code decoding: **M** (Magnetic stripe)
- Prefix: **N**

It may be important to set an adequate "Ignore card".

3.1.5 Active cards for "hands-free"

Cards with ISO dimensions but with a thickness of about 3 mm, with lithium battery, produced by Cotag.
The following basic settings are recommended

- Type of code decoding: **W** (Wiegand)
- Prefix: **T**

3.1.6 Active vehicle Tags

Vehicle Tags with lithium battery, produced by Cotag, to be fitted under the vehicle.
The following basic settings are recommended

- Type of code decoding: **W** (Wiegand)
- Prefix: **T**

It may be important to set an adequate "Ignore card".

3.1.7 Cards and transponders type HID

See the specific paragraph for details and descriptions.
The following basic settings are recommended

- Type of code decoding: **W** (Wiegand)
- Prefix: **T**

3.2 Magnetic Stripe

The most common readers with a Magnetic-Stripe output are:

- both swipe and insertion type readers of magnetic cards
- passive proximity card readers
- Telepass type readers for vehicle applications
- Radio-control receivers (see Cardin company)

To provide full information, please note that, on request, Contag Controllers can be programmed for a Magnetic-Stripe output.
However, this is used only in very special cases. Programming of Tag Cotags becomes very complicated and compatibility with the usual Tag Cotags is lost.

The Control unit requires a Magnetic-Stripe code with a maximum of 37 characters + 3 control characters according to ABA standard.
A magnetic reader on ISO2 track requiring a maximum of 37 characters + 3, must be connected.
These are the control characters:
Start sentinel (B)hex ..... START ..... located at the beginning of the useful code
End sentinel (F)hex ..... STOP ..... located at the end of the useful code
LRC ..... LRC ..... located after the End sentinel (Longitudinal Control of Redundancy)
The string must, therefore, be of the following type: START + DATA (max. 37) + STOP + LRC

- If codes are found to be correct order during decoding, reading is "**normal**".
- If codes are found to be in reverse order (first LRC + STOP + DATA + START) during decoding, reading is "**reverse**".

3.3 Wiegand

Readers with Wiegand output are typically:

- readers of active cards (Cotag controller)
- HID readers (see paragraph)

The Control unit requires a 32 bit or 26 bit Wiegand code. If it receives 26 bits, it switches automatically to recognise the HID type transponders. The relevant paragraph describes how the data are interpreted.
If 32 bits are received, they are seen as 16 + 16 bits.

- The first 16 bits are the Installation Code in binary code. They are transformed into decimal code and take on values from 00000 to 65535.
- The second 16 bits are the Card Code in binary code. They are transformed into decimal code and take on values from 00000 to 65535

With "**T**" Prefix (see below)
The card is communicated to the Wincontrol program (or internally managed if Stand-alone) as a 10-character string - the sum of the 5 Installation Codes and the 5 Card Codes.

With "**N**" Prefix (see below)
The card is communicated to the Wincontrol program (or internally managed if Stand-alone) as a 10-character string - the sum of the 5 Installation Codes and the 5 Card Codes.
So, only the Prefix changes, but the rest of the code is the same.

3.4 "T" and "N" Prefix

It is very important to set the prefix because it involves the way of interpreting the read codes.
This interpretation involves both the Wincontrol program and the Control unit.

Each reader can be individually set on the Control unit. Example: reader **A** Prefix "**T**" and reader **B** Prefix "**N**".

It is called "prefix" because the most evident effect of programming is that the first character of the string containing the data of the read card sent to the Controller is a "T" or an "N" (e.g.: NØØ426ØØØ32678?……..).
N.B.: We inform the person monitoring communications with the "Terminal Emulation" function, that in the sent string, the "T" was replaced some time ago by a "K", but the prefix is still referred to as a "T" prefix for the sake of continuity.

With "**T**" Prefix

The Control unit:

- reads all codes on the card (e.g.: ØØ426ØØØ32678)
- identifies the Installation Code and the Card Code in precise positions (fixed for all cards).
- transforms the Installation Code into a 5-character number with zeros at the beginning if necessary (e.g.: ØØ426)
- transforms the Card Code into a 5-character number with zeros at the beginning if necessary (e.g.: ØØ32)
- unites the two codes into a single 10-character string (e.g.: ØØ426ØØØ32)
- sends the data of the read card with the "T" prefix + 10 characters.
- characters that are not Installation Code or Card Code (e.g.: 678) are ignored.

Whatever is analysing the code (the Centre if On-line or the Control unit if Stand-alone), if it detects that the prefix is "T":

- expects 10 numeric characters after the prefix and sees them as two groups of 5 + 5
- interprets the first 5 as the Installation Code
- interprets the second 5 as the Card Code
- scans the internal Card archive to evaluate if a Card with that Number and that Installation Code exists.
- if it does not exist, the card is not valid – if it does exist, the different validity tests applicable are run (time-bands, etc.).

To exploit this function, one must be able to program the cards:
it must be possible to identify on the code one part that clearly refers to the Card Code, and another to the Installation Code.
This can be done with :

- new magnetic cards specifically programmed for that installation.
- active Cotag cards

Prefix "T" is preferable since it is easier to manage.

With "**N**" Prefix

The Control unit:

- reads all codes on the card (e.g.: 4567ØØ324ØØØ256781987)
- if specified, the initial part and final part of the code (only in special applications) are cut in order to compose a "useful code" (ØØ324ØØØ25678)
- sends data with the "N" prefix + useful code (NØØ324ØØØ25678).

Whatever is analysing the code (the Centre if On-line or the Control unit if Stand-alone), if it detects that the prefix is "N":

- scans the internal Card archive to evaluate if a Card with that code in the "Character sequence Association" field exists.
- if it does not exist, the card is not valid; if it does exist, the Card Code and the Installation Code are extracted from it for identification; then the necessary validity tests are run (time-bands, etc.)
- if valid, the card is archived in the historical archive with the Card Code and Installation Code obtained from the Card archive, and not from the numbers read on the card ("Character sequence Association" field).
- if it does not exist but communicates with the Wincontrol program, it is, in any event, sent to enable autoreading.

It is clear that the archives must have been prepared beforehand, so that the corresponding code was loaded in the "Character sequence Association" field card by card: the Autoreading function is used.
N.B.: it is usually advisable to have a reader (and relevant Control unit) near the PC with the Wincontrol program to facilitate Autoreading.

Most common use.

The “N” prefix is used in the following cases:
- passive proximity cards (they have a univocal, random, non re-programmable code)
- existing cards from whose code it is impossible to identify if one part of the code clearly refers to a Card Code and another part to the Installation Code.

3.5 Code control settings

The contents of the following paragraph refer only to a reader with a Magnetic-Stripe output.
The length of the read code can be up to 37 characters.

With “**T**” Prefix

The Control unit executes the following controls and operations:
- it checks if the length of the read code is longer than “Minimum length” (default 1 but programmable from 1 to 37). If it is shorter, it aborts that card reading operation.
- it checks if the length of the read code is shorter than “Maximum length” (default 37 but programmable from 1 to 37). If it is longer, it aborts that card reading operation.
- goes to the position indicated by “Start of Installation Code (default 1 but programmable from 1 to 37) and takes as many characters as specified in “Installation Code Length” (default 5 but programmable from 1 to 5)
- the read characters are, in any case, converted into a 5-character string with zeros at the beginning on the left, and form the Installation Code (e.g.: if I.C. length = 3 and the 3 characters are 234, they become ØØ234)
- goes to the position indicated by “Start of Card Code (default 6 but programmable from 1 to 37) and takes as many characters as specified in “Card Code Length” (default 5 but programmable from 1 to 5)
- the read characters are, in any case, converted into a 5-character string with zeros at the beginning on the left, and form the Installation Code (e.g.: if C.C. length. = 4 and the 3 characters are Ø519, they become ØØ519)
- the 10 obtained characters (5 Installation Code characters + 5 Card Code characters) are sent to the Controller with prefix “T”.

Example

1	2	3	4	5	6	7	8	9	10	11	12	...	32	33	34	35	36	37	• Position
0	**0**	**7**	**3**	**5**	**0**	**0**	**3**	**8**	**2**	**9**	**5**	**...**	**x**	**x**	**x**	**x**	**x**	**x**	• **Numeric Code**

N.B.: there is nothing from position 13 to 37, i.e. the card is 12 characters long.
Minimum Card length = 12 Maximum Card length = 12
Start of Installation code = 1 Installation code Length = 5
Start of Card code = 6 Card code Length = 5
Reading result:
Minimum length test OK: there must be at least 12 characters
Maximum length test OK: there must be 12 characters at the most
ØØ735 ➔ Installation Code
ØØ382 ➔ Card Code
Sent as TØØ735ØØ382 (in Terminal Emulation one would actually see K;ØØ735ØØ382?011832190599 A)
Note that with any other card length (e.g.: 13 characters) the reading would be aborted.

With “**N**” Prefix

The Control unit executes the following controls and operations:
- it checks if the length of the read code is longer than “Minimum length” (default 1 but programmable from 1 to 37). If it is shorter, it aborts that card reading operation.
- it checks if the length of the read code is shorter than “Maximum length” (default 37 but programmable from 1 to 37). If it is longer, it aborts that card reading operation.
- goes to the position indicated by “Start of character sequence” (default 1 but programmable from 1 to 37) and eliminates all characters preceding the specified position.
- goes to the position indicated by “End of character sequence” (default 37 but programmable from 1 to 37) and eliminates all characters following the specified position.
- the characters thus obtained are sent to the Controller with prefix “N”.

Example

1	2	3	4	5	6	7	8	9	10	11	12	...	32	33	34	35	36	37	• Position
0	**0**	**7**	**3**	**5**	**0**	**0**	**3**	**8**	**2**	**9**	**5**	**...**	**x**	**x**	**x**	**x**	**x**	**x**	• **Numeric Code**

N.B.: there is nothing from position 13 to 37, i.e. the code is 12 characters long.

Minimum Card length = 1 Maximum Card length = 37
Start of character sequence =3 End of character sequence = 10
Reading result:
Minimum length test OK: there must be at least 1 character
Maximum length test OK: there must be 37 characters at the most
735ØØ382 ➔ Card Code
Sent as N735ØØ382 (in Terminal Emulation one would actually see N735ØØ382?011832190599 A)

3.6 LRC Control

This strictly concerns the Magnetic-Stripe input.
A control character known as LRC is saved on the magnetic stripe just after the useful characters. It depends strictly on all the other characters on the magnetic stripe.
During reading, the Control unit checks that all characters read on the magnetic strip generate an LRC equal to the one actually read on the strip itself – otherwise, the card reading is ignored because this means a problem had occurred.
However, insertion readers may read only 60% of the magnetic strip, and it may therefore be impossible to insert the card sufficiently to read up to the LRC.
The reading would normally be aborted, but if the first characters of the card can be read, the following can be done:

- disable LRC control
- cut off the final part of the code, using the "End of character sequence" (for "N" Prefix only").

3.7 Ignore Cards

The "Ignore Card" function can be enabled separately on reader A and reader B.
The time needing to elapse so that the same card can be read again on the same reader can be set for each reader (A and B).
Note the following:

- the system compares the code of the last card read with that of the penultimate one: if it is the same, it checks if the ignore card time has elapsed.
- if ignore card time is set at zero (as a pre-setting) the function is de-activated.
- a second card that is different from the first is read immediately even if the set ignore card time has not elapsed.
- time is expressed in seconds, and can be in the range from 0 to 255.
- the Tenths/Seconds flag does not influence this function

Example: time is 10 seconds. The terminal remembers the last recognised card and, if it is presented within 10 seconds, it is ignored.
Use: Remember, that for readers of insertion magnetic cards, about 10 seconds are recommended so that the cards are not read as they are extracted. For proximity readers too (e.g.: Cotag), 10 to 15 seconds should be set for personal cards and about 60 seconds for vehicle Tags read by a Loop ahead of a gate.

3.7.1 Ignore all cards

If this function is enabled, the reader (A and B separately) can be disabled for the entire "ignore card" time.
In fact, all the read cards are ignored during this period, and not just the last one.

3.7.2 Ignore card on other reader

If set to this mode, when a reader reads a card, in addition to verifying its own ignore card function, it is also verifies if that card is subjected to the ignore card function on the other reader. If it is, it is ignored.
It simultaneously involves both reader "A" and "B".
There are a multiplicity of uses, but the typical case is of two vehicle readers of active cards (with reading distances of over 1 metre) both located on the pavement separating the entry and exit lanes.
As these readers read well in reverse direction too, the danger is that a card may be read involuntarily for a second time by the other reader too.
However, if the "Ignore card on other reader" function is enabled, the second reading will be ignored.
E.g.: a card is read on A and the relevant ignore time is 10 sec.: this card is ignored for 10 seconds on both A and B.
If the r**eader A "Total" card Ignore** function were simultaneously activated on A, only the card read on A would be ignored on B, and not all cards.

3.7.3 Card Multiple ignore on other reader

This function is similar to the previous one. These are the main differences:

- can be enabled separately on **reader A** and on **reader B**
- the latest reading is ignored if it coincides with one of the last 10 readings of the other reader.
- the last 10 valid readings are saved on each reader
- each reading is subject to timeout – when this time has elapsed, it is removed from the "last 10 readings" memory.
- an ignore time (timeout) other than the "Ignore card" time can be set.

This function is typically used when using readers with a very broad reading range (e.g.: Transponder in GigaHz) where the risk cannot be entirely avoided whereby a Transponder is involuntarily read by the other reader after being correctly read by its own reader (e.g.: entry/exit lanes of a car park).
To **enable** this function, just set a time other than zero; this time will be timeout associated with each reading. Maximum selection time is 255 seconds equal to 4 min. and 15 sec.
To **disable**, set a time equal to zero.
To **enable** the function for **reader A** means:

- every valid reading on **reader A** is saved along with the associated timeout period.
- up to **10** readings are saved for **reader A**
- if a valid reading is effected when there are already 10 pending readings (i.e. timeout not yet elapsed), the last reading is written over the oldest.
- the timeout period of each stored reading is reduced every second; if it reaches zero, that reading will be removed from the memory of the "last 10 readings" and the card in question will again be readable on **B**.
- every reading on **reader B** is ignored if it has already been effected on **A** within the timeout period.

To **enable** the function for **reader B** means:

- every valid reading on **reader B** is saved along with the associated timeout period.
- up to **10** readings maximum are saved for **reader B**
- if a valid reading is effected when there are already 10 pending readings (i.e. timeout not yet elapsed), the last reading is written over the oldest.
- the timeout period of each stored reading is reduced every second; if it reaches zero, that reading will be removed from the memory of the "last 10 readings" and the card in question will again be readable on **A**.
- every reading on **reader A** is ignored if it has already been effected on **B** within the timeout period.

N.B.: the memory of the last 10 + 10 valid readings is a function valid in both **On-line** and **Stand-alone** modes and does not interfere with the memory of the events to be downloaded if the Control unit is operating in **Stand-alone** mode.

3.8 Disable Time-Band test

If the Control unit is in "Stand-alone" mode, during the card validity check, it is evaluated if the card is in the Time-Band.
If there is no need for a given panel to run this test (e.g.: an external vehicle gateway), control can be disabled so that all cards presented at that panel are valid round-the-clock for 365 days a year.
Obviously, the other tests, such as Card archive, "Access Level", etc. remain enabled.
To disable the Time-Band test:

- run the Wincontrol program, accessing the "Time-Band test" item of the Panel Archive, and remove the tick-mark.
- effect a download operation toward Control unit.

If in "On-line" mode, it is the Wincontrol program that, during its validation process, ignores the "Time-Band" test, whereas, in "Stand-Alone" mode, it is ignored by the Control unit.

3.9 Disabling the "Installation code" test

The "Installation code" test can be disabled if necessary.
This can be done only if in **"T" Prefix**. In fact, there is no sense in talking about "Installation Code" with the "N" Prefix ("character sequence association").
This function should operate in the following cases:

- Stand-alone
- On-line
- Magnetic-stripe entry
- Wiegand entry

If in "Stand-alone" mode, any “Installation Code” read on the card is acceptable for card validity. When transit data are communicated to the Controller, the “Installation Code” programmed on the Control unit will be used.
If in "On-line" mode, transit data will be communicated to the Controller, using the “Installation Code” programmed on the Control unit and not that read on the card.

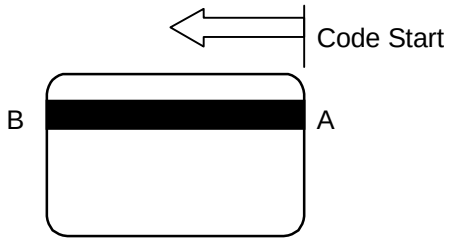
3.10 Display and communication settings

There are four types of card reading:

1. “**normal**” reading on reader **A** (**NA**)
2. “**reverse**” reading on reader **A** (**RA**)
3. “**normal**” reading on reader **B** (**NB**)
4. “**reverse**” reading on reader **B** (**RB**)

A reading is considered “normal” when the code sent by the reading head starts from the first character and ends with the last.
A reading is considered “reverse” when the code sent by the reading head starts from the last character and ends with the first.
The “normal” and “reverse” reading concept is typical of magnetic card readers:

- if **swipe-type** the following apply to readers:
  - “normal” is a reading from right to left (🡐)
  - “reverse” is from left to right 🡒
- if **insertion-type**:
  - “normal” reading during insertion
  - “reverse” reading during extraction.



The figure shows a card viewed from the magnetic-strip side.
The code begins with A and finishes with B. If ends a little before B if the string is shorter than 37 characters (ISO2).

- If the reading magnetic head in the card reader first meets point A and then B, the reading is “normal”.
- If the reading magnetic head in the card reader first meets point B and then A, the reading is “reverse”.

“Double” passive proximity readers have a “normal” Magnetic-Stripe output and a “reverse” type output according to the antenna which read the card.
Readers of Cotag active cards (and Wiegand in general too) only have the “normal” output.
The distinctions we have just made are important for setting the behaviour of the Control unit for card reading.
The following can be defined for each of the 4 previous cases (NA, RA, NB, RB):

- reason (from 0 to 9)
- entry/exit/neutral (E, U, ‘ ‘)
- on-display message for a valid card (6 characters. e.g.: Output).

3.10.1 Reasons

When a card is read (or downloaded later if it was validated while in "Stand-alone" mode), the Control unit is also informed of the reason associated with the type of reading effected.
The reason is typically used for Presence Detection applications. In fact, the Wincontrol program does no more than file the information in the historical archive.
If the 8-Hour module is built-in, the information serves to reconstruct if transit was an entry or an exit.
The information is reconstructed by associating certain reasons with Entry and other with Exit. (E.g.: 1=Entry; 2=Exit).
Warning: do not use the Entry / Exit indications of the next paragraph for this purpose.
A card reading is typically communicated complete with the corresponding Reason.
For Presence Detection purposes, one can enable the facility for typing in a Reason before the card is read.
In that case, one can type in just one numeric key from 0 to 9.

3.10.2 Entry/Exit/Neutral

When a card is read (or downloaded later if it was validated while in "Stand-alone" mode), the Wincontrol program is also informed whether the type of reading effected also serves for "Present Count" or for "Antipassback" purposes.
The information in question can take on one of the three following values:

1. " " (a space) Neutral (default value)
2. "**E**" Entry
3. "**U**" Exit

If the Wincontrol program receives an "E" transaction

- if Antipassback is active, the card is considered valid only if it is "Absent".
- the program forces the card to "Present" and increases the "present counter" by one unit.

If the Wincontrol program receives an "U" transaction

- if Antipassback is active, the card is considered valid only if it is "Present" – however the "Exit always valid" flag can still be activated in the Wincontrol program to prevent any problems at the exit.
- the program forces the card to "Absent" and reduces the "present counter" by one unit.

If the Wincontrol program receives a transaction with " " (space)

- it does not concern the preceding tests. The present counter is not influenced. This is the default situation.

N.B.: **DO NOT** use this function to define **Entries** and **Exits** under **Presence Detection**!!! Use the "Reasons".

If necessary, the communication of any selected E/U ("Do not close-queue E/U") can be disabled.
In this case, both the On-line valid Card communications and the delayed communication of a valid card while in "Stand-alone" mode occur by queuing a " " (space), i.e. Neutral.
In situations when the "Exit always valid" function is used in the Wincontrol program, reference is made to a reading containing a "U". The information is transferred to the Control unit by a data download.

3.10.3 Word on display

You can set a string with a **maximum** of **6** characters. If the **card is valid**, this string will be shown on the display of the relevant terminal (either on A or on B) – on the bottom left line. There are 4 possible words respectively associated with the readings NA, RA, NB and RB. The Control unit displays 6 spaces as the default setting.
Therefore, words such as "**ENTRY"** or "**EXIT"** can be displayed according to how the card was read.
The words can be personalised, by using keyboard Programming (at the "DISPLAY TEXTS" item) or by using the Wincontrol program.

3.11 "PINSOST" function (Substitution Pin)

With this function, the user does not require any card for opening and clock-in .
It is active in "Stand-alone" mode too.
The function must be programmed in the "Panels" Archive of the Wincontrol program.

- write the following in the "Mode" field: PINSOST
- effect a download operation to Control unit

To make use of the function (e.g.: because the user has forgotten his card) proceed as follows:

1. type in the card number (clear numbers on the first line)
2. press the "#" key (the cursor moves to the second line)
3. type in the PIN associated with that card (asterisks are shown on the screen instead of numbers)
4. press the 'E' Enter key

At this point the information is sent to the Wincontrol program if the Control unit is "On-line" or processed locally if in "Stand-Alone" mode.

- You cannot type "Reasons".
- The reading cannot be indicated as a "Reverse" reading.

E.g.: card 19 associated with PIN 429602.
You must type: 19#429602E where '#' is the gate key and 'E' is the Enter key on the keypad.

3.11.1 Display of PINSOST on a line

This function was introduced to maintain compatibility with the previous Control units.
If enabled, the characters are all displayed on the first line and all are not clear characters.

3.12 "PIN" Function

This function makes card routing more secure at some gateways.
It is active in "Stand-alone" mode too.
The function must be programmed in the "Panels" Archive of the Wincontrol program.

- write the following in the "Mode" field: the word PIN
- effect a download operation to Control unit

Procedure for effecting a card reading operation :

1. type the PIN code associated with the card + Enter key
2. Have the card read

The order of execution of the two operations can be **reversed**.
A longer time period that the one specified under the "Initialisations" – "System configuration" in the "Special transits timeout" field (preset at 16 seconds) of the Wincontrol program must not elapse between one stage and the next.
When the PIN is being typed in, asterisks should appear instead of numbers on the screen.
If this does not happen, this means programming was incorrect.

E.g.: card 19 associated with PIN 429602.

- You must type: 429602E where E is the Enter key
- Have the card read

If the function is enabled, a "Reason" can be typed before the card is read, but an asterisk is displayed.

3.12.1 Cards not requiring a PIN

Card by card definition is possible – in the relevant archive – on whether a card is subject to PIN or not.
In all panels defined as PIN, cards not subject to PIN must not type it.

3.12.2 Suspension of Pin on a time-band

In the Wincontrol program , the "Suspend PIN" function can be activated for each panel if in a Time-Band.
A maximum of three Bands can be specified. Zero reset all three to disable the function. If Time-Bands are shown, this is the procedure:

- if you are in at least one of the three Time-Bands, card reading only with PIN is sufficient.
- if you are not in any of the three Time-Bands, card reading with PIN is necessary

At download from the Centre, the information is transferred to the Control unit, which continues behaving as described even if in "Stand-alone" mode.

3.13 Justifications

An up-to-date version of the Wincontrol program (see manual) is required – a version able to recognise, manage and archive the justifications.
Justifications are needed mainly for Presence Detection and for the Industrial Control processes.
If you enable this function, you can type in up to 6 numeric keys before having the card read.
The complete information (card number, date/time, reason, E/U, reader A/B, justification) is immediately communicated to the Centre if "On-line" or is saved in the Control unit's historical archive if in "Stand-Alone" mode, to wait to be downloaded when communication is restored.
In "Stand-Alone" mode, enabling the "Justifications" means reducing the card memory's capacity by about one third. See the table in the characteristics section.
Settings
The "**Justifications"** function must be enabled. This means that the amount of storable events is **reduced**.
You can set both the **minimum number** of keys for typing in as well as the **maximum number**.
The first is a number from 1 to 6. For the maximum number, load in a number not lower than the first number (if the minimum number is 3, it is of no use setting a maximum number of 2 or 1: you should set from 3 to 6).

If the **minimum number** is **0** and the "Justifications" are enabled, you may type in before the card is read. If no key is typed, the transmitted Justification consists of 6 "spaces". You may type in a number of keys equal to the quantity set in **maximum number**.
If the **minimum number** is in the range **1** to **6**, you must type in at least that number of keys before the card is read, otherwise the card will be ignored and the operation will be aborted.
The **maximum number** is optional and can be a number from 1 to 6 and, in any event, equal to or higher than the **minimum number**.
N.B.: the following rule **must be** complied with: enable or disable the function **only if there are no events** in the memory to be downloaded. We therefore advise you to connect to the Control unit on-line and check if any events are still being sent from it - if it is set via Modem, download the data and prevent card reading until the settings are made.
If you enable the "Justifications", you obtain the following:
1. communication of Justifications is enabled
2. if in "Stand-Alone" mode, any event archived in the internal historical archive occupies more memory space than the standard level, even if no Justification was typed in: Card readings, Alarms/Restore ops. of an input (if the function is enabled).
3. the information is shown at the centre of the display, while markers on the line below help you type in: "-" refers to an obligatory digit; while "^" refers to an optional digit.

Operational procedure
1. **Type the obligatory keys**, and the optional keys if necessary. **N.B.: do not type other keys (e.g.: "E")**.
2. Have the card read **while** typed–in Justification is shown.

A separator can be included in the Justification (but not at the beginning); this is the "#" key, which will be communicated as a "," (comma) and will be indicated as such in the Reports of the Wincontrol program.
Each character in the communicated Justifications can be: a number from "0" to "9", a comma, "," , and a space " ".
The space " " cannot be typed in but is inserted by the Control unit to fill the string and take it up to 6 characters (they are added on at the left).
If you type in more than the specified maximum number of keys, the display resets.
However, the Justification will always be communicated with a length of 6 characters: if you press less than 6 keys, the number of necessary spaces " " to complete the 6 digits, will be added on at the left.
If a card is read without typing any keys, this means that a Justification of 6 spaces " " (only if the minimum number is zero) is queued.
Card readings effected while the specified min/max number had **not** been typed in, will be ignored.
To make it obligatory to type in 6 keys (for example), 6 must be set as both the maximum and minimum number.
When the "Justifications" are enabled, the possibility of typing in the Reason becomes meaningless: however, the Reason associated with the type of reading will be queued. (E.g.: if the reading took place on reader B by swiping from left to right, the Reason specified in RB will apply, i.e. Reverse B).
With the "Justifications", the Reason essentially takes on the meaning of the swiping "Direction".
N.B.: we advise you **not to** enable **PIN** or **PINSOST** simultaneously with the **Justifications.**
If the "Justifications" are enabled, the executed setting will apply **simultaneously** to both reader **A** and reader **B**.

B Justification on A
In certain applications, it may be convenient to have just one keyboard/display connected as reader A and a second reading head connected to B but without a keyboard/display. The aim is to type in the "Justifications" on the only keyboard (A) provided, and to read the card either on reader A or B.
This can be done by just activating the function.

Display of the '#' key
This setting can be made separately for reader A and reader B.
This setting also influences the displays in PIN and PINSOST (if on one line). However, it becomes important in connection with the Justifications.
If the "#" key is not typed in at the beginning of the line, it is usually shown as a hyphen ("-").
In some applications, a comma (",") is shown, as in the case of numbers with decimals.
In other applications, the gate comma sign ("#") is displayed.
In yet others, a space (" ") is shown.

3.14 Antipassback

This function only has any meaning if the Control unit makes decisions locally regarding the validity of the card, i.e. if it is in "Stand-Alone" or "Always Stand-Alone" mode.
Antipassback means preventing two or more consecutive transits in the same direction. E.g.: two Entries or two Exits.
Therefore the purpose of this function is:

- prevent Entry by a card which is already Present
- prevent Exit by a card which is already Absent

Procedure to enable this function:

- enable **Antipassback**
- enable at least one type of reading (NA, RA, NB, RB) with direction "**E**" (entry)
- enable at least one type of reading (NA, RA, NB, RB) with direction "**U**" (exit)

If using a magnetic swipe reader, you can enable "E" in one direction and "U" in the other.
A card download from the Centre updates the local Absent/Present situation of each and takes priority over the previous local situation. Therefore, if the Control unit considers card No. 520 as Present and then receives a data download in which that card is Absent, the Control unit considers card No. 520 as Absent from that moment on.
Downloading to the Centre of the gradually accumulated events after a period in Stand-alone mode, helps to update the Absent/Present situation in the Centre's archive, but only if the E/U queuings are enabled.
The Antipassback function can be enabled together with the Count.

Exit always valid

In practice, if the Antipassback is operating at Entry only for security reasons, the "Exit always OK" function can be enabled, i.e. exit always valid.
N.B.: in this case, an **exiting** card will be considered valid (direction associated with "U"):

- whether it is absent or present
- whether in the Time-Band or off Time-Band

In other words, when the "Exit always valid" function is enabled, exit controls are less severe.

3.15 Count

When the Count function is enabled, the Control unit can locally count the number of present within a certain area. The entry/exit readers must necessarily be connected to the Control unit.
Procedure to enable this function:

- enable **Count**
- it is preferable to force the Control unit into "**Always Stand-alone**" mode
- enable at least one type of reading (NA, RA, NB, RB) with direction "**E**" (entry)
- enable at least one type of reading (NA, RA, NB, RB) with direction "**U**" (exit)
- set the number of "**Max Present**" which, in the case of a car park, would be its capacity.
- in the case of a **car park**, also consult the "**Fraud events**" paragraph as fraud events affect the count.
- if "**Transits**" are enabled, it is the "**Valid transit**" event that updates the count and not just the "Card reading".

If using a magnetic swipe reader, you can enable "E" in one direction and "U" in the other.
The number of Present is obtained by scanning the local card archive, and totalling how many are "Present".
If the number of Present is greater than or equal to the "Max Present", the **Excess** situation is in action, i.e. **new entries are denied .**

Excess

When the count of Present reaches the "**Max Present**" number, the Control unit goes into **Excess** mode.
This influences the following activations:

- ❖ **OUT1-AUX** is activated
- ❖ signalling **LED3** is activated

In Excess mode:
Entries are denied

1) But **Exits** are accepted

Vacant/Full indicator board

As **OUT1-AUX** is activated in the **Excess** mode, this output (open-collector) can be used for commanding the indicator board showing the Vacant/Full status.
N.B.: the "**Max Present**" number must be other than zero.

Alarm Control unit

You may sometimes wish to activate an alarm Control unit when all are recorded as Absent, and then re-activate it when at least one is Present. All you have to do is:

- set "**Max Present**" = 0

When all are absent, **OUT1-AUX**: is activated.
Only one present is necessary for **OUT1-AUX**: to be disabled.

The Count function can be enabled together with Antipassback.

3.15.1 Privileged card management (Vip and SuperVip)

By enabling the “Privileged card management” function, you can sub-divide the cards into three groups, using the “Presence Detection Code” as the characterising element.
It operates in “Stand-alone” mode only, whereas “Count” is active and the Control unit is in “Excess” mode.
New Entries are normally prevented in this condition. However, it may be necessary for someone to enter nevertheless: enable the function and associate the “Presence Detection Code” to the cards, as follows.

- **normal** cards (“Presence Detection Code” = **0**) – Do not pass through and Entry if in Excess mode
- **VIP** cards (“Presence Detection Code” = **1** or **2**) –Pass through an Entry even if in Excess mode. They are included in the Count
- **SuperVIP** cards (“Presence Detection Code” = **3**) – Pass through and Entry even if in Excess mode. They are not included in the Count.

To be precise, the specification “included in count” or “not included in count” is applied even if the system is not in Excess mode.
This function is available as from Ver. 01.02.03.

4 Miscellaneous functions

4.1 Tenths / Seconds

On the Control unit, the count of activation times of:

1. OUT1 output (for valid card, for activation associated with an entry or for external control)
2. OUT2 output (for valid card, for activation associated with an entry or for external control)
3. OUT3 output (for valid card, for activation associated with an entry or for external control)
4. OUT4 output (for valid card, for activation associated with an entry or for external control)

is in "**Tenths**" of a second or in "**Seconds**".
The number sent by the Wincontrol program is, in any case, saved in Tenths according to the following:

- if in Tenths at download, the time is saved as it was received (e.g.: 10 tenths)
- if in Seconds, it is multiplied by 10 and saved (e.g.: if 10 was received, it is saved as 100 tenths).

For activation, the corresponding time (in tenths) is loaded from the memory and executed.
As the default setting is in "Tenths", remember to indicate activation times in tenths in the Wincontrol program (e.g.: activation time of 2 seconds = 20 tenths).
N.B.: remember that if a Control unit is in "Tenths" and is then switched over to "seconds" or vice versa, a **new data download** from Controller is necessary in order to adapt the times of the four outputs.
N.B.: by programming from the keyboard, you can type in long times, even if they are in Tenths. In seconds, only if operating from the Centre.
Remember that a subsequent download from the Centre will overwrite any manual settings.

4.2 Programmable inputs logic

Inputs IN1, IN2, IN3, IN4, IN5, IN6, IN7 are switchable in the function logic.
This makes it possible to connect both normally open (NO) and normally closed (NC) contacts.
This function is described better in the "Connections" paragraph.
You can use either the Wincontrol program or "keyboard programming":

4.3 Event memory block

If the connection between the Control unit and the Centre is interrupted, the events can be saved and then downloaded to the PC as soon as the communication is re-established.
It the available memory is completely full, the most recent data erase the oldest. However, operation continues. In some applications, it is preferable not to lose any data-item at the cost of blocking readings of any further cards.
If the "Event memory block" is activated, when the memory is full, this is what happens:

- further card readings are ignored
- a message is shown reporting the status and prompting you to download to the PC.

4.4 Baud Rate change

You can change the Baud Rate of the serial port used for communication with the PC (COM1) and of the Auxiliary serial port (COM2).
The COM1 port has two electric interfaces to be used according to the application:

1. RS232 galvanically separated on terminal board COM1-RS232
2. RS485 galvanically separated on terminal board COM1-RS485

It is clear that if one of the two interfaces is used, the other must not be used otherwise electric conflicts will occur.
The two following specific settings are available:

- one for the Control unit set as /**P**
- one for the Control unit set as /**T**

If in /**P**, the preset speed is **57,600, S, 8, 1.**
Generally, changing the Baud Rate is not necessary.
Other standard speeds can be selected with "keyboard programming".
If you select another speed (e.g.: 19,200), the polling port is initialised at 19,200, S, 8, 1.

The same port is used if the Control unit is set as "via Modem" (/T).
If in /**T**, the preset speed is **19,200, N, 8, 1.**
Other standard speeds can be selected with "keyboard programming".
If you select another speed (e.g.: 9,600), the Modem port is initialised at 9,600, N, 8, 1.

The COM2-Aux port has one interface only – the RS232. The following can be changed: Baud rate, parity, No. of bits, No. of stop bits.
The set Baud rate comes into force only if at least one function is activated on the COM2-Aux, otherwise it is 57600, N, 8, 1.

4.5 Extended displays and keyboards

The Control unit usually manages this type of display 16 x 2, i.e. 16 characters on 2 lines.
The standard keyboard has 16 keys (a matrix of 4 x 4 keys)
The Control unit's board is however also used in devices using a larger than standard display and keyboard. This display is a 20 x 4, i.e. 20 characters on 4 lines and the keyboard has 22 keys.
In this case, to ensure management, set as follows:

- the display of reader **A** is **20 x 4** instead of the default of **16 x 2**.
- the display of reader **B** is **20 x 4** instead of the default of **16 x 2**.
- the keyboard of reader **A** is **extended** (22 keys) instead of the default of the **standard** keyboard (16 keys)
- the keyboard of reader **B** is **extended** (22 keys) instead of the default of the **standard** keyboard (16 keys)

These settings must be made in keyboard programming.
When using the 20 x 4 display, line 2 and 3 (in the centre) show the same words as the 16 x 2 display.
Lines 1 and 4 can be, instead, freely used for showing fixed texts, which can be customised.
Customising can be effected from the Centre or by using the "DISPLAY TEXTS" item in keyboard programming.
To write a text quickly, it is more convenient to work from the Centre, as follows:

- Select the item "Communication" – "EEPROM reading/writing Panels".
- Select the Panel number to be initialised.
- If you have to set Line 1 in the "Commands" field (low down), type in **+1025sz;xxxxxxxxxxxxxxxxxxxx** where xxxxxxxxxxxxxxxxxxxx are a maximum of alphanumeric 20 character to be viewed.
- If you have to set Line 4 in the "Commands" field (low down), type in **+1046sz;xxxxxxxxxxxxxxxxxxxx** where xxxxxxxxxxxxxxxxxxxx are a maximum of alphanumeric 20 character to be viewed.

Keyboard programming is more laborious since you cannot directly type in the characters on the keyboard.
After an EEPROM clean-up (e.g.: update of firmware with a new version sufficient to cause the EEPROM to be cleaned), any "customised" text is erased from both the lines and is replaced by the default text.
The "customised" text must therefore be re-programmed.

4.6 "HID" (Hughes) readers

As we said in the paragraph on the interfaces of the Wiegand readers, the passive card readers produced by "HID" (Hughes) can be interfaced.
These readers are:

1. Prox Pro
2. Mini Prox

The following passive cards can be used:

1. DuoProx II
2. Iso Prox II
3. Prox Card

The reader outputs must be 26-bit "WIEGANDs" (the 37-bit version is not tolerated). The format must be:

E	I	I	I	I	I	I	I	I	I	I	I	I	T	T	T	T	T	T	T	T	T	T	T	T	O
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26

- E is the parity bit – Even – in the first half of the message
- I are the 12 bits of the Facility Code (Installation Code); the most significant bits are transmitted first
- T are the 12 bits of the Card Number (Card Code); the most significant bits are transmitted first
- O is the parity bit – Odd – in the second half of the message

The Control unit must be

- put into **Wiegand** mode

If the data output is as expected, the Control unit will automatically recognise the 26-bit format.
Connect a 5-pole cable as follows (the terminals on the Control unit are indicated first, then those on the "HID"):

+12 (pin5) → DC+	GND (pin6) → GND	D1 (pin7) → DATA1	D0 (pin8) → DATA0	BUZZ (pin9) → Beep

4.7 Crosspoint Emulation

This function is linked to the old systems with passive readers producing a 9-character Magnetic-stripe string.
Instead, a present day passive card reader produces a 14-character Magnetic-stripe string.

Clearly, if there are old readers (Crosspoint) and new readers in the system, the same card will produce two different codes according to the reading point.
In this case, the Control unit can be programmed to transform the 14-character codes of the new readers into the 9-character codes which would have been produced by a Crosspoint reader.
This function can be activated separately on reader A and B.
Programming is not possible through a specific menu item, but is done by directly writing in a location of the EEPROM memory.

- To **activate** the function on reader **A** → write **255** in location **302** (write 0 to disable)
- To **activate** the function on reader **B** → write **255** in location **303** (write 0 to disable)

No card heading is discriminated: all 14 character length codes are converted to 9 characters and the other characters are therefore ignored (e.g.: if a code is of 13 character length, it remains 13).
Activating Crosspoint Emulation is useful only if a "**Magnetic-stripe**" reader input is set and if in "**Prefix N**".
This function is available as from Ver. 01.02.04.

5 Miscellaneous management

5.1 Reader A on one gateway and Reader B on another

5.1.1 Introduction

The Control unit and the Wincontrol program are **not** structured for completely separate management of reader A and reader B installed on different gateways. However, accepting some limitations, a single Control unit (complete with 2 readers: A and B) can be used to manage access of 2 separate gateways. A reader must be installed at each gateway: e.g reader A at door A, and reader B at door B.

N.B.: there are several **limitations**. These are the main ones:

- If the "Access level" is used in the card archive of the Wincontrol program, readers A and B cannot be distinguished if the same Access level is applied. If, however, "Access Groups" are used, readers A and B can be made to behave differently.
- non optimised reports for each reader

Also remember that:

- the function is also active in "Stand-alone" mode
- do not use "Action messages" for a valid card if it has to come into force on one reader only
- do not use "Action messages" or "**invalid** card activation ops." if they are to be executed for one reader only

5.2 Disabling reader A and reader B

Readers A and B can be disabled individually or both together.
The function can be programmed and is usually non operational.
The following alternatives are possible for reader A:

Reader A disabled for Inp. 5 ?		
No (always enabled)	If Inp. 5 enabled (closed)	If Inp. 5 disabled (open)
1	2	3

The number from 1 to 3 matches the one on the display in keyboard programming.

The following alternatives are possible for reader B:

Reader B disabled for Inp. 6 ?		
No (always enabled)	If Inp. 6 enabled (closed)	If Inp. 6 disabled (open)
1	2	3

The number from 1 to 3 matches the one on the display in keyboard programming.

Therefore, the behaviour of the readers is as follows, according to the selected option:

	Inp. 5 (6) disabled	Inp. 5 (6) enabled
1	Reader A (B) enabled	Reader A (B) enabled
2	Reader A (B) enabled	Reader A (B) disabled
3	Reader A (B) disabled	Reader A (B) enabled

Remember that the input logic can always be reversed to adapt its behaviour.

Text displayed before or after card reading

When an input is disabled, a message is shown on the display to inform the user.
However, in some situations, one may prefer to view the disabling message only after the card was read. In that case, the "View after disable" function must be programmed.
It can be set individually for A and B.

Authorised activation cards only

Usually, when an input is disabled, all cards are invalid.
However, in some situations, it may be necessary to authorise certain cards to be validated: the "Activation" or "Activation only" cards:
In this case, the "Authorise activation cards" function must be programmed.
It can be set individually for A and B.

In order to know if a card is an activation card, the Control unit obtains the information from the local archive in the RAM: An updated data download must have been made into the archive, and the card must be in the archive.

Message on display
A number from 1 to 3 can be set for each reader (A/B) identifying which text is to be shown when the reader is disabled (depending on the state of the associated input). The following messages correspond to each number:

- If 0, the **date / time** are shown
- if 1, **"VEHICLE NOT PRESENT"** is shown
- If 2, **"READER DISABLED"** is shown
- If 3, "**ALARM SYSTEM ON"** is shown

Message No. 0 is normally set (date / time).
Message No. 3 can be selected for interfacing with the Alarm Control units (see further below).
A different Message can be set for A and B.

5.3 Interfacing with Alarm Control Units

Two inputs of the Control unit can be used for interfacing the relay output of an Alarm Control unit in order to disable both the readers (A, B or A + B) and display status .
Use the "Disable reader A and B" function.
N.B.: to disable both reader A and B, you must simultaneously operate on inputs 5 and 6.
Another **Text** to be viewed must be set only for disabling purposes.
The display **Text** to be shown is No. 3.
This must be done for each reader involved (A and B).

5.4 " A Door management", "B Door management", "Single-door management"

"Door management" means controlling any unauthorised openings of the gateway and use of a door-opener push-button to exit.
The Control unit is supplied in "Normal" mode but can be switched to "**A Door management"** or "**B Door management**"
This function therefore refers to both reader A and reader B.
The following can be managed

- One door only with reader A
- One door only with reader B
- Two distinct doors with two readers (one per door)
- One door only with two readers ("**Single-door management**"). This is the mode tolerated by the previous cards.

As concerns the **exits** for A and B door openers, what was said in the paragraph "Reader A on one gateway and Reader B on the other" applies in this case too.
If the "**A Door management** " mode is set, inputs IN1 and IN2 of the Control unit can no longer be used as general alarm inputs, but must be connected to the following devices:

Input **1**	**A door status** microswitch. If normally closed when the door is closed, the logic must be reversed.
Input **2**	Internal push-button for activating **Gateway A door opener**

If the "**B Door management**" mode is set, inputs IN3 and IN4 of the Control unit can no longer be used as general alarm inputs, but must be connected to the following devices:

Input **3**	**B door status** microswitch. If normally closed when the door is closed, the logic must be reversed.
Input **4**	Internal push-button for activating **Gateway B door opener**

The input for the door status microswitch (IN1 or IN3) remains an alarm input, but can be disabled for a certain time when one of the following conditions occurs:

- transit of a valid card (even if using the substitution PIN)
- pressing the internal door-opener push-button connected to input 2 (if A) or input 4 (if B).

The disabling time can be set in the range from 1 sec to 255 sec max.
The preset value is 15 sec.
Pressing the door-opener key causes the following:

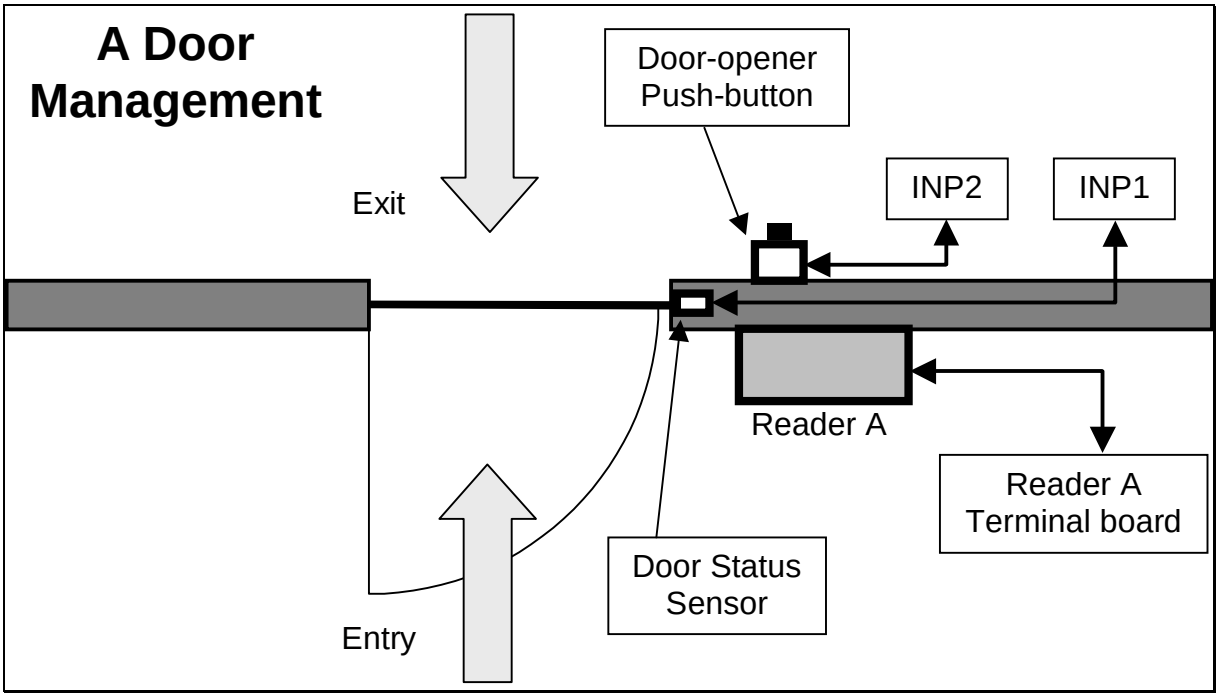
- all "valid card" relays are energised, and may be **filtered** for reader **A** or **B,**
- the corresponding **input is disabled** for the described time.

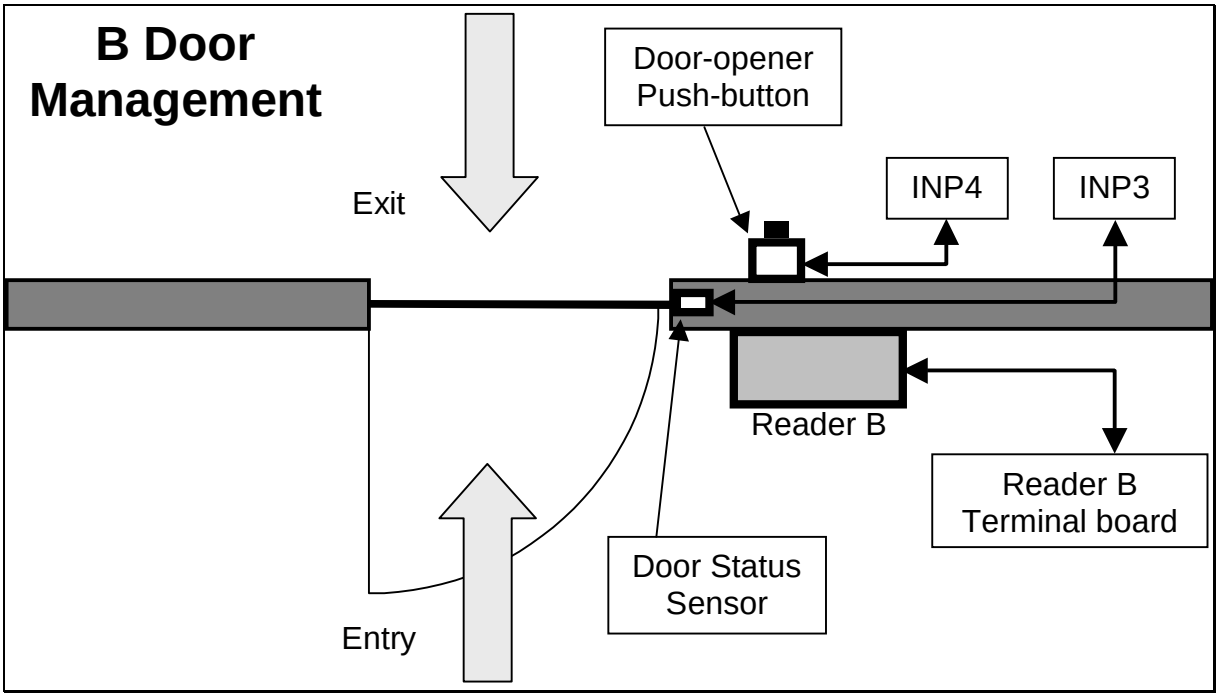
The **door status input is immediately enabled** when the door is re-closed (even if this happens before the time elapses): so if you open the door with the push-button, you will disable door status for 15 sec, but if you open the door, pass through and close it within 8 seconds, a new opening of the door would generate an alarm.

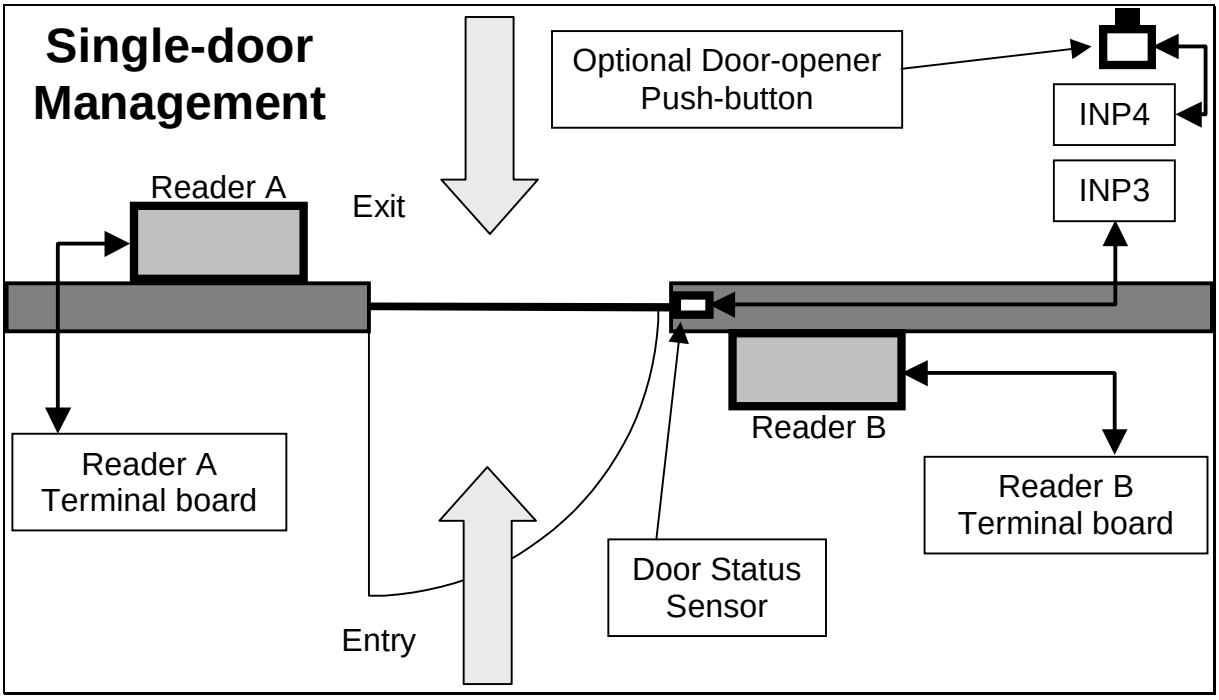
To sum up, presenting a valid card or pressing the door-opener push-button cause the same standard activating operations to be executed.
Obviously, what we said does not concern

- Activation cards where only the standard activating operations remain in common
- Activation **only** cards: a set of activating operations are enabled with the card, another set with the door-opener push-button
- Action messages

The following diagrams show the possible situations.







N.B.: in "Single-door management" both "A Door management" and "B Door management" must be **disabled**.
Note that the fundamental difference between "A Door management" and "Single-door management" is: in the first case, a valid card on A will disable INP1 for a certain time; in the second case, it will disable INP3 for a certain time.

5.5 Activation operations by Time-Band

One or more Time-Bands in the archive can be associated with each of the outputs (OUT1 – OUT4). These outputs will be activated for the entire validity period of the Time-Band.
Up to 5 activation operations can be specified by Time-Band: obviously, the same output can be indicated several times to generate more complex Time-Bands.
Setting must be done in the Wincontrol program using **"Archive" – "Panels" + data download to panels.**

In this function, it is advisable to use outputs not utilised for other purposes, even if there are exceptions: for example, an output for door-opener maintained continuously open at certain hours of the day (by using electric strikes), and activated with valid Cards at other hours.
Application examples: light on, buzzers, closing an entry in order to generate timed activation operations, etc.
Management is performed by the Control unit whether it is in "On-line" or "Stand-alone mode", but the data must **first** be **downloaded** from the PC with the Wincontrol program to the Control unit.
If an output (e.g.: OUT1) is active as in Time-Band, it stays active even if it is activated for a few seconds by a valid Card or by a timed activation operation associated with an input.
If the disabling of an input is associated with an output (time 255), that output stays active.
Any Reset (e.g.: tripping of the system's Watchdog) causes the disabling of the outputs in question, equal to the Reset time plus about one second.

5.6 Buzzer function

This function is possible only if the Wincontrol program in use tolerates that function (i.e. it must be quite recent).
It can operate even if the Control unit is in "Stand-alone" mode. As concerns this function, the connection to the Wincontrol program serves only for downloading the programming from the PC and for keeping time aligned.
The purpose of the function is to periodically activate one of the outputs in order to sound a buzzer or a bell.
A free output among the four available outputs must be identified and dedicated to this function.
An activation time and one or more archived Time-Bands (12 max) can be associated with one of the Control unit's outputs (OUT1 – OUT4).
This output will be activated for the time set, both at the beginning and at the end of the Time-Band validity period.
If end of Time-Band activation is not required, the same time must be indicated for both the beginning and end of the band.
Up to 12 Time-Bands can be associated, and, therefore, up to 24 activations are available (this depends on how they are set).
One output only and one activation time only can be specified.
Activation time is in tenths (default value) or in seconds confirming to the Control unit's setting.

- if set in Tenths, times of up to 25.3 seconds can be set.
- if set in Seconds, times of up to 253 seconds can be set.

Remember that setting in Tenths or in Seconds also influences valid card activation times.
The data must be set in the "Panel" archive of the Wincontrol program: to make the data executive, the data must be downloaded to the panel in question.
A different configuration can be set for each Panel.
To ensure that activation is executed, the Control unit must be operating while the first minute of Band validity begins, and likewise for the end of validity minute.

E.g.: the output is OUT4, time is 200, Control unit set in Tenths, and the associated Time-Bands are No. 7 and No.12, which in this example are "No. 7: 8.00 – 12,00 of M, T, W, T, F, S" and "No. 12: 14.00 – 18.00 of M, T, W, T, F".
Result:

Monday:	8 am for 20 sec.	12 noon for 20 sec.	2 pm for 20 sec.	6 pm for 20 sec
Tuesday:	8 am for 20 sec.	12 noon for 20 sec.	2 pm for 20 sec.	6 pm for 20 sec
Wednesday:	8 am for 20 sec	12 noon for 20 sec	2 pm for 20 sec	6 pm for 20 sec
Thursday:	8 am for 20 sec	12 noon for 20 sec	2 pm for 20 sec	6 pm for 20 sec
Friday:	8 am for 20 sec	12 noon for 20 sec	2 pm for 20 sec	6 pm for 20 sec
Saturday:	8 am for 20 sec	12 noon for 20 sec		
Sunday:	nil			
Holidays:	nil			

5.7 Alternative uses of OUT1-AUX

The "OUT1-AUX" output is on the auxiliary connector and is the only Open-collector output.
This output has no explicit associated functions, unless the local "Count" is enabled in the Control unit.
In that case, OUT1-AUX signals the Excess status (see relevant paragraph).
If the count is not enabled, this output can be used for other applications.
To do this, output **No. 5** must be specified in the masks of the Wincontrol program.
The command received from the Centre and activating "OUT1-AUX" can be:

1. permanent activation (command with **time = 254**)

2. timed activation (command with **time from 1 to 253**)
3. disabling of output (command with **time = 255**): this operates whether a permanent or timed activation is enabled.

N.B.: certain functions activate OUT1-AUX only if "On-line" (e.g.: activation associated with inputs). Therefore, we recommend using this output only if no other outputs are free and only after checking the effective functionality under all conditions.

5.8 Inputs Engagement and Disengagement times

The "Engagement" and "Disengagement" time can be individually set for each input from Inp1 to Inp6. This is not possible for the Tamper (Inp7)
"**Engagement" time** means that one can filter and ignore the **activations** of the input in question, which are shorter than the set time.
"**Disengagement" time** means that one can filter and ignore the **de-activations** of the input in question, which are shorter than the set time.
Times are in **Tenths** and can be set in the range from **0 to 255 tenths**. (from 0 a to 25.5 seconds)
They are very useful in the "**Parking**" function for filtering the status of the Presence and Transit **loops** because they often do not have a stable output if the vehicle is moving. Finally, it serves to prevent false commutations of the logic status.
E.g.: input 5 is at rest, Engagement time is 15 and Disengagement time is 40.
If the input is activated for less than 15 tenths, it is not considered in alarm status. It goes into alarm status, if it is activated for more than 15 tenths.
At this point, Inp5 is in alarm status. If the input is de-activated for less than 40 tenths, it is not considered to be restored.
It will be considered restored, if Inp5 is de-activated for more than 40 tenths.

5.9 On-line printing of valid cards

Each reader can be enabled for on-line printing of a line containing the data of the card reading that was just performed.
This is the function to enable: "**Reader A card print OK**" and **"Reader B card print OK**":
Printing is performed only if the card is **valid**.
It does not matter if the Control unit is in On-line or Stand-alone mode.
An 80-column **RS232 serial** printer must be connected to the Control unit.
The printer must be connected to the Control unit's **COM2** port.
Default communication occurs at **9600, N, 8, 1**. Other values can be set.
The printed line contains the following information:

- date/time
- Card No.
- On-line / Stand-alone (i.e. if the validity decision is made by the Wincontrol program or by the local Control unit).
- Panel No.
- Reader (A/B)
- Normal / Reverse (of significance with magnetic readers)
- The programmable 6-character text shown on the display
- Reason (from 0 to 9)
- Direction (E, U, neutral)
- Justifications (only if enabled)

N.B.:

- according to the **language** used, the **record format** could be slightly **different** (i.e. some fields could be a little longer or shorter).
- the **record format changes** according to whether the **Justifications** are enabled or not.

If you have to update the firmware with printing activated, we advise you to put the Control unit into keyboard programming mode.

Connections

Control unit side (COM2 – Aux port)		Printer side
Pin on terminal M13	**Name**	**Name**
1	**TX**	**RX**
5	**GND**	**GND**

For pin numbering on the printer, consult the relevant manual.
N.B.: It may be necessary to make jumpers (between RTS and CTS) on the printer-side connector to ensure printing is executed. Consult the manual: especially the "handshake" mechanism on the serial port.
N.B.:

- if **at least** one Print is activated, the Baud rate of COM2 becomes as set (default **9.600, N, 8, 1**)
- if no function is activated for COM2, the settings are **57.600, N, 8, 1**.
- if you access the "keyboard programming" mode, the COM2 settings are **57.600, N, 8, 1**.
- If you have to do a "**firmware update"** with this function activated, we advise you to put the Control unit into **"keyboard programming"** mode.
- these settings are effective if there is a **reset** (or power-down + power-up) after it was activated. This is important if the setting is done from the Wincontrol program.
- this function is not in contrast with the other uses of the COM2 port, and, therefore, they can be activated. However there will be connection problems if Print and Indicator Board are simultaneously enabled.

5.10 Indicator Board Management on COM2

By activating this function, you can send a string of characters from the Wincontrol program and arrange for the same string to be duplicated at the COM2 output. If a Backlit indicator board is connected to COM2, texts can be displayed at your discretion. In the Wincontrol program, texts can be composed in the **"Areas count"** section.
When composing the text, remember that:

- to obtain a **binary 0** character, type in "**0**", i.e. backslash + ASCII zero.
- to obtain the \ **backslash** character, type in "\\", i.e. backslash + backslash

N.B.:

- if **at least** this function is activated, the Baud rate of COM2 becomes as set (default **9.600, N, 8, 1**)
- if no function is activated for COM2, the settings are **57.600, N, 8, 1**.
- if you access the "keyboard programming" mode, the COM2 settings are **57.600, N, 8, 1**.
- If you have to do a "**firmware update"** with this function activated, we advise you to put the Control unit into **"keyboard programming"** mode.
- these settings are effective if there is a **reset** (or power-down + power-up) after it was activated. This is important if the setting is done from the Wincontrol program.
- this function is not in contrast with the other uses of the COM2 port, and, therefore, they can be activated. However there will be connection problems if Print and Indicator Board are simultaneously enabled.

5.11 Barcode Reader on COM2

By activating this function, you can connect a barcode card reader with "Barcode" on the COM2 serial port.
This function has the following limitations:

- **only** one reader can be connected to each Control unit
- this reader is seen as if it were **reader A**
- all **settings** valid for reader **A** apply
- the traditional reader B (e.g.: magnetic or proximity) can be connected to the relevant terminal board B
- the traditional reader A (e.g.: magnetic or proximity) can also be connected to the relevant terminal board A but there could be conflicts or problems during simultaneous readings with Barcode: use it only if you are certain that the user is using either one type of reader or another (e.g.: has only optical or passive cards)
- the Barcode reader must have an **RS232** output
- the string at Output must have a maximum length of **37** useful characters + **CR** (or **LF** or both) as terminator
- **non numeric** characters are transformed into "**0**" ASCII (the **numeric** characters are from ASCII "**0**" to "**9**)
- both Prefix T and Prefix N are acceptable for reader A
- it does not necessarily have to be a reader of cards with Barcodes: it can be a generic device with RS232 output, with any type of reading

Connections

Control unit side (COM2 – Aux port)		Barcode reader side
Pin on terminal M13	**Name**	**Name**
2	**RX**	**TX**
5	**GND**	**GND**

For Pin numbering on the Barcode reader, consult the relevant manual.

Use with Prefix "T" on reader A
If the string received on COM2 and sent by the Barcode reader is "**0178900324987**"+ LF, the Control unit considers that Card No. **324** was read with Installation Code N° **1789** (if the default settings remain).

Use with Prefix "N" on reader A
If the string received on COM2 and sent by the Barcode reader is "**0178900324987**"+ LF, the Control unit considers that Card with the Character sequence "**0178900324987**" (if the default settings remain, i.e. without cutting off the sequence).

N.B.:

- if **at least** this function is activated, the Baud rate of COM2 becomes as set (default **9.600, N, 8, 1**)
- if no function is activated for COM2, the settings are **57.600, N, 8, 1**.
- if you access the "keyboard programming" mode, the Com2 settings are **57.600, N, 8, 1**.
- If you have to do a "**firmware update"** with this function activated, we advise you to put the Control unit into **"keyboard programming"** mode.
- these settings are effective if there is a **reset** (or power-down + power-up) after it was activated. This is important if the setting is done from the Wincontrol program.
- this function is not in contrast with the other uses of the COM2 port, and, therefore, they can be activated

6 Board settings

The board enables the user to select some operating modes and keep data associated with them in a non-volatile EEPROM memory (therefore, data are not lost following a reset, a power cut, or disconnection from the lithium battery). These parameters can be set as follows:

- By "Keyboard programming", providing a keyboard/display is supplied.
- from the Controller if the panel to be set is on-line. In that case, the following setting procedure are available:
  1. using the Wincontrol program and selecting the "Panel Archive" item
  2. using the Wincontrol program and selecting the "Communication" – "Programming On-line Panels" item.
  3. using the Wincontrol program and selecting the "Communication" – "Initialising On-line Panel" item.
  4. using the Wincontrol program and selecting the "Communication" – "Panel Configuration" item.
  5. using the Wincontrol program and selecting the "Communication" – "EEPROM reading/writing" item.
  6. using the Wincontrol program and selecting the "Communication" – "Terminal Emulation" item.

N.B.: each of the above procedures makes it possible to execute specific settings that cannot be accessed by the others with the exception of procedures 4) and 5) which enable setting all panel parameters.
The difference is that procedures 1), 2), 3) and 4) suggest screen masks to be used for setting the data-item and then downloading it automatically to the Control unit, in the correct location, without any action by the operator.
Procedures 5) and 6) make it possible to write any data item in any EEPROM location. Consequently this potentially dangerous operation must be effected by duly instructed personnel, and only by writing in the locations for which the relevant documentation is available.
Settings effected from the Wincontrol program are explained in the relevant manual or on on-line Help.
The settings, which can be made from "Keyboard programming" are almost all the possible settings.
Using the keyboard, you can program any location directly, writing the value in decimals.
This virtually allows any setting although the operating procedure cannot always be described in the manual. In special cases, consult the Manufacturer on what to do.

6.1 Keyboard programming

If only the A keyboard/display is supplied, or both readers: programming is possible on A only.

Keyboard programming can be accessed in two ways:

1. using key pair **7** and **9**
2. using the 3-pin **CN7** connector

Activating the programming mode with keys 7 and 9

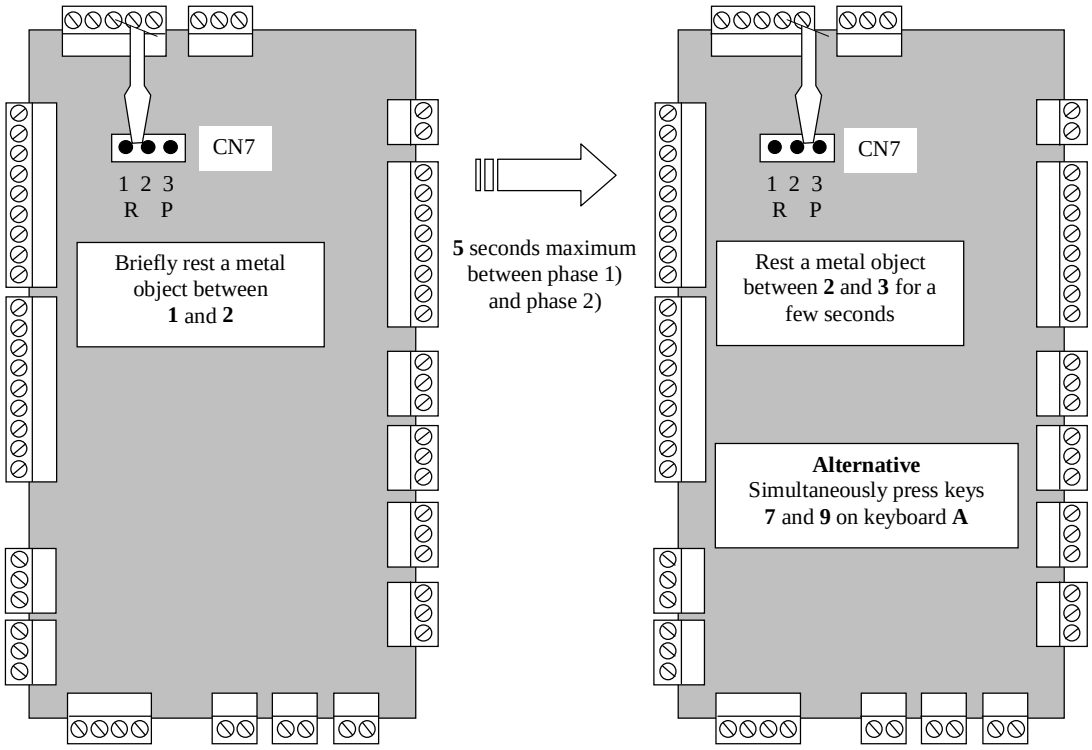
Procedure by powering down the board:

- cut power to the panel and then power-up again (or reset the board by short circuiting Pins **1** – **2** of **CN7**)
- simultaneously press keys "**7**" and "**9**" of the keyboard (you have 5 sec to perform this operation)
- release keys "**7**" and "**9**" when you hear the confirmation beeps.

Activating the programming mode with the CN7 connector (see following figure):

Procedure by powering down the board:

- cut power to the panel and then power-up again (or reset the board by short circuiting Pins **1** – **2** of **CN7**)
- create a brief short-circuit between Pins **2** – **3** of **CN7** (you have 5 sec to perform this operation)
- release the connection when you hear the confirmation beeps.

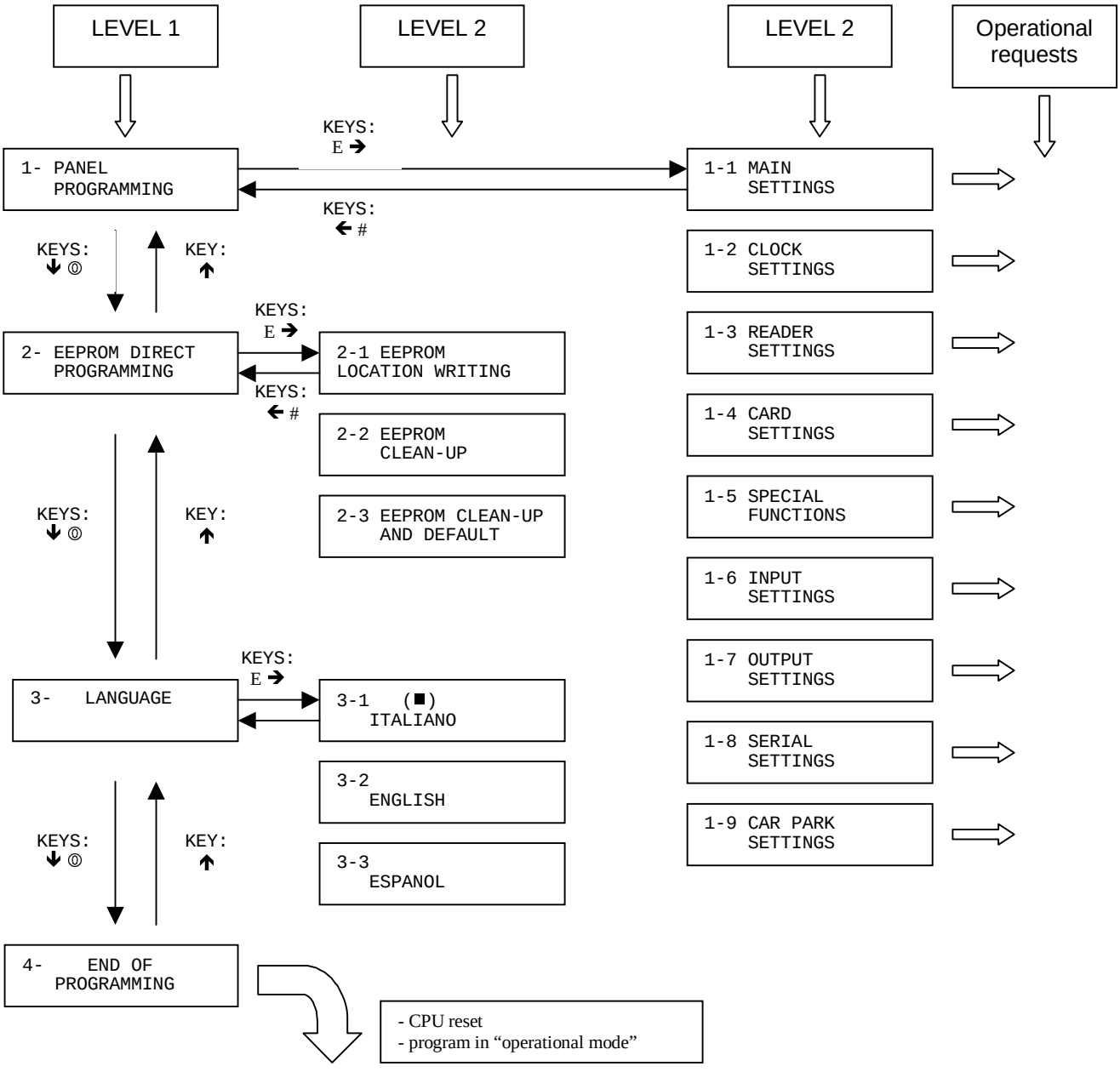


Phase 1):
Reset

Phase 2):
Programming

N.B.: the following rules apply for moving around the masks of the "menus":

- **the menu is split into various levels (currently 2)**
- **use keys "↓" or "0" to sequentially view the menu items**
- **to view menu items in reverse sequence, use key "↑" (but only if provided on the keyboard)**
- **when you have finished scanning a level, resume from the first of the same level; when scanning in reverse order, resume from the last. A longer beep informs you that the cycle has finished**
- **to confirm a menu item (changing over to the level further on the right), use keys "E" or "→"**
- **to return to a level further on the left (toward level 1) use key "#" or "←"; this can be done from any point you may have reached**
- **to retrieve a menu item more quickly, you can use the associated numeric key (e.g.: you are located on item "1-1 MAIN SETTINGS ", by pressing key "7" you view item "1-7 OUTPUT SETTINGS ")**
- **when you have reached the level further on the right (typically level 2), the "operational request" begins – this can include one item or several items in sequence. To return to the previous level (cancelling the operation in progress), press key "#" or "←" (number being typed in is signalled, the key "#" cancels the typing-in in progress; if you press the key "#" a second time, you return to the previous level)**



OPERATIONAL REQUESTS

N.B.: in the following programming masks:

- **for "yes/no" selections or where 2, 3, or 4 alternatives are required, the flashing cursor suggests the current selection**
- **for requests with 3-digit values, the current value is displayed in round brackets**
- **for requests with 5-digit values, the current value is not displayed**
- **when typing-in any number, there is not need to add zeros in front of the number**
- **press the key "E" to confirm current value and move on to the next mask**

6.1.1 Menu "1- Panel programming" – "1-1 Main settings"

```
PANEL NUMBER
1-255  ___ (  1)
```

To change the current number of the panel (shown in brackets), type in the new number and confirm with key 'E'. Type in a number in the range 1 to 255 that does not already exist on the network.

```
INST. CODE
(    1) _____
```

To change the current installation code of the panel (shown in brackets), type in the new code and press key 'E'.

```
3=POLLING /P
1=MODEM-TERM. /T
```

To modify the protocol mode on the COM1 serial port.
Set Polling (/P) if connected to "Spacenet" in RS485 or connected directly to the PC in RS232.
Set "Modem / Terminal Server" (/T) if a Modem or a Terminal Server on LAN Ethernet is coupled to the Control unit.

```
JUSTIFICATIONS
 1=Yes    3=No
```

This is used to enable the facility for typing in a Justification before the card is read. Before modifying, you are prompted to confirm Yes/No. Remember that this setting changes the maximum number of storable events.

```
PRES.DETCT.MODE
 1=Yes    3=No
```

This is used to enable the "Presence Detection Mode".
N.B.: activating this function also entails:

- ❖ the "Max number of Cards" is reduced to a very low level to leave more space in memory for the Events.
- ❖ the Prefix of readers A and B is changed to "T" (so be careful if the Prefix had been "N")

Disabling the Presence Detection Mode implies taking the "Max number of Cards" to the default value (1500).

The current value of "Maximum number of cards" and "Maximum number of events" are displayed for a few seconds, then the mask for setting a new value for "Maximum number of cards" appears.

```
NUMBER OF CARDS
( 1500) _____
```

If a new value is set, the Control unit calculates and shows for a few seconds the new "Maximum number of events" and, before implementing the modification, asks for confirmation – Yes/No.

```
HIST.ARCH.BLOCK.
 1=Yes     3=No
```

If the events historical archive is full, a new event usually cancels the oldest. If the historical archive is blocked, this does not happen. Obviously, when this happens, card readings are disabled and a video message reports that a data-download is necessary to free memory space.

```
A TRANSIT MANGM.
 10 Yes   3=No
```

Enables and disables management of "Transits" on reader A. **Attention:** it enables communication of a second event, after the card reading, which may be relevant to counts, antipassback, etc. **Consult the relevant paragraph**

```
B TRANSIT MANGM.
```

```
 1=Yes      3=No
```

Enables and disables management of "Transits" on reader B. **Attention:** it enables communication of a second event, after the card reading, which may be relevant to counts, antipassback, etc. **Consult the relevant paragraph**

```
ON-LINE/ST-ALONE
  1)   2)   3)
```

Three alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. These are the functions:

1) standard behaviour: i.e. if there is a communication (Polling) it goes into On-line mode, otherwise it operates in Stand-alone (decisions made by the Control unit)
2) forcing the "Always Stand-alone" mode (i.e. decisions made by the Control unit even if it is on-line with the Centre).
3) forcing the "Always On-line" mode (i.e. the Control unit does not make decisions by delegates them to the Centre).

```
ANTIPASSBACK
 1=Yes    3=No
```

This is used to enable **local** antipassback in "Stand-alone" mode.

```
COUNT ENABLING
 1=Yes    3=No
```

This is used to enable **local** count in "Stand-alone" mode.

```
MAX PRESENT
(00100) _____
```

Defines the threshold for comparison with local counter in order to establish if in "Excess" mode

```
3 = SECONDS
1 = TENTHS
```

In "**Tenths"**, the times communicated by the Wincontrol program are interpreted in tenths of a second and, therefore, activation times are possible from 1 to 253 tenths, i.e. from 0.1 to 25.3 sec.
In "**Seconds"**, the received number is interpreted in seconds thus enabling times from 1 to 253 (4 min and 13 sec).
Remember that the times from 254 to 255 are reserved for defining:

- 254 output always enabled
- 255 disables an output

Important: after changing over from "Tenths" to "Seconds" or vice versa, do a new data **download**.

```
RTC CALIBRATION
0-255  ___ (  4)
```

This is used for setting a number enabling fine adjustment of the local clock (Real Time Clock).
Consult the manufacturer before making any changes.

```
3=DISPL.16x2 (A)
1=DISPL.20x4 (A)
```

This is used for setting the type of display connected to reader A. This can be a display with 16 characters on 2 lines (default) or with 20 characters on 4 lines.

```
3=DISPL.16x2 (B)
1=DISPL.20x4 (B)
```

This is used for setting the type of display connected to reader B. This can be a display with 16 characters on 2 lines (default) or with 20 characters on 4 lines.

```
3=NORM.KEYB.(A)
1=EXT.KEYB. (A)
```

This is used for setting of the two keyboard options for reader A: standard keyboard (16 keys on a 4 x 4 matrix) or extended keyboard (22 keys).

```
3=NORM.KEYB.(B)
1=EXT.KEYB. (B)
```

This is used for setting of the two keyboard options for reader B: standard keyboard (16 keys on a 4 x 4 matrix) or extended keyboard (22 keys).

```
DISPLAY TEXTS
 1=Yes    3=No
```

If you select 1), you access a sequence of masks where you can view and modify the following texts:

1. text for valid card with NA reading (6 characters max)
2. text for valid card with RA reading (6 characters max)
3. text for valid card with NB reading (6 characters max)
4. text for valid card with RB reading (6 characters max)
5. line 1 of display if 20 x 4 set (20 characters max)
6. line 4 of display if 20 x 4 set (20 characters max)

To modify the texts, the keys are used as follows:

- left arrow key (🡐) +or "**4**" shift the cursor to the left along the text; it stops at the beginning of the string
- right arrow key (🡒) +or "**6**" shift the cursor to the right along the text; it stops at the end of the string
- up arrow key (🡑) +or "**2**" change in upward sequence the alphabet characters, numbers and punctuation signs
- down arrow key (🡓) +or "**8**" change in downward sequence the alphabet characters, numbers and punctuation signs
- key "**0**" inserts the line terminator in the current position; if pressed again, it restores the previous character.
- key "**1**" provides the following in sequence: number "1", letter "A", and letter "a". This is to speed up pointing at the character you are looking for; you then use the arrows to select the right character
- key "**3**" provides the following in sequence: letter "A", letter "G", letter "R", letter "a", letter "g" and letter "r". This is to speed up pointing at the character you are looking for; you then use the arrows to select the right character
- key "**5**" is used to take the cursor to the beginning of the line
- key "**7**" is used for inserting a "space" in the current position; the following characters shift by one position to the right ; there is no insertion if the line is already full (permissible number of characters reached)
- key "**9**" is used for removing the character pointed at by the cursor; the following characters shift by one position to the left; the text length is reduced by one unit.
- key "**F1**" (applies only to an extended keyboard, a 20 x 4 display and modification of line 4) is used to provide – in sequence – some of the most important texts used for Clock-in (e.g.: "Exit", "Entry", "Exit --> <--Entry", etc.)
- key "#" is used to exit and move on to the next text; any modifications made to the test are not saved.
- key "**E**" (Enter) is used for confirming the modifications made and for moving on to the next text; it can be used for displaying various texts in sequence

6.1.2 Menu "1- Panel programming" – "1-2 Clock settings"

N.B.: in the following masks, each modification (e.g.: seconds) comes into force individually when the "E" confirming key is pressed.

```
   TIME
0-23   ___ ( 10)
```

Setting the hour on the local clock

```
  MINUTES
0-59   ___ ( 29)
```

Setting the minutes on the local clock

```
  SECONDS
0-59   ___ ( 47)
```

Setting the seconds on the local clock

```
1=MON     7=SUN
 1-7   ___ (  4)
```

Setting the day of the week on the local clock (1=Monday; 2=Tuesday; 3=Wednesday; 4=Thursday; 5=Friday; 6=Saturday; 7=Sunday).

```
  DAY
1-31   ___ ( 10)
```

Setting the day on the local clock

```
  MONTH
1-12   ___ (  5)
```

Setting the month on the local clock

```
  YEAR
0-99   ___ (  1)
```

Setting the year on the local clock (001 = 2001)

6.1.3 Menu "1- Panel programming" – "1-3 Reader settings"

```
3=MAG-STRIPE (A)
1=WIEGAND    (A)
```

Sets the type of data format expected by reader A. The Magnetic-Stripe format (default setting) is used with readers of magnetic cards with a TTL output, with passive proximity readers, and with vehicle transponder readers in Gigahertz for long distances, etc.
The Wiegand format is typically used with active card readers or with HID readers (see paragraph)

```
3=MAG-STRIPE (B)
1=WIEGAND    (B)
```

Sets the type of data format expected by reader B. The Magnetic-Stripe format (default setting) is used with readers of magnetic cards with a TTL output, with passive proximity readers, and with vehicle transponder readers in Gigahertz for long distances, etc.
The Wiegand format is typically used with active card readers or with HID readers (see paragraph)

```
3=PREFIX A: N
1=PREFIX A: T
```

Press key "1" to set prefix T (default) for reader A.
Press key "3" to set prefix N for reader A. Remember that, for passive proximity readers, prefix N must be activated.

```
3=PREFIX B: N
1=PREFIX B: T
```

Press key "1" to set prefix T (default) for reader B.
Press key "3" to set prefix N for reader B. Remember that, for passive proximity readers, prefix N must be activated.

```
IGNORE CARD A
0-255  ___ (  0)
```

Set the ignore time of the last card read on reader A.

```
IGNORE CARD B
0-255  ___ (  0)
```

Set the ignore time of the last card read on reader B.

```
TOTAL IGNORE A
 1=Yes    3=No
```

If you enable this function, all cards – and not just the last read card – will be ignored on reader A during card ignore time.

```
TOTAL IGNORE B
 1=Yes    3=No
```

If you enable this function, all cards – and not just the last read card – will be ignored on reader B during card ignore time.

```
IGN.BETW.A AND B
 1=Yes    3=No
```

This function is activated to accept a card and also check if it was recently read by the other reader: in other words, the card will be ignored if it was the last one read on the other reader and the relevant ignore time has not elapsed.

```
MULTIPLE IGN.A-B
0-255  ___ (  0)
```

Enables the "Multiple Ignore" function on reader B for the last readings on A.

```
MULTIPLE IGN.B-A
0-255  ___ (  0)
```

Enables the "Multiple Ignore" function on reader A for the last readings on B.

```
READER A LRC
 1=Yes    3=No
```

Refers to reader A. To correctly decode a Magnetic-Stripe code, the control character (LRC) must be checked. LRC control must be disabled for insertion readers reading only 60% of the magnetic stripe coupled to cards with over 18 characters. For further information, see the relevant paragraph.

```
READER B LRC
 1=Yes    3=No
```

Refers to reader B. To correctly decode a Magnetic-Stripe code, the control character (LRC) must be checked. LRC control must be disabled for insertion readers reading only 60% of the magnetic stripe coupled to cards with over 18 characters. For further information, see the relevant paragraph.

```
READER A ENABLED
  1)   2)   3)
```

Three alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. These are the functions:

1) standard behaviour; i.e. reader A always enabled
2) reader A is enabled only if Input 5 is disabled; if Input 5 is activated, reader A is disabled.
3) reader A is enabled only if Input 5 is enabled; if Input 5 is disabled, reader A is enabled (inverse logic compared to point 2)

```
READER B ENABLED
  1)   2)   3)
```

Three alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. These are the functions:

1) standard behaviour; i.e. reader B always enabled
2) reader B is enabled only if Input 6 is disabled; if Input 6 is activated, reader A is disabled.
3) reader B is enabled only if Input 6 is enabled; if Input 6 is disabled, reader A is enabled (inverse logic compared to point 2)

```
TEXT FOR IN5
 0-3   ___ (  0)
```

In the previous programming operations, enabling reader A depended on the status of input 5. However, with this mask, you can modify the type of message which appears on the display when reader A is disabled.

- The standard view with date/time (even with the reader disabled) is associated with text No. 0.
- The "VEHICLE NOT PRESENT" message is associated with text No.1.
- The "READER DISABLED" message is associated with text No.2 (default value).
- The "ALARM SYSTEM ON" message is associated with text No.3.

```
TEXT FOR IN6
 0-3   ___ (  0)
```

In the previous programming operations, enabling reader B depended on the status of input 6. However, with this mask, you can modify the type of message which appears on the display when reader B is disabled.

- The standard view with date/time (even with the reader disabled) is associated with text No.0.
- The “VEHICLE NOT PRESENT” message is associated with text No.1.
- The “READER DISABLED” message is associated with text No.2 (default value).
- The “ALARM SYSTEM ON” message is associated with text No.3

```
ACT.CARD A ONLY
 1=Yes    3=No
```

By activating this function, the “Activation” or “Activation only” cards are enabled to avoid being ignored by reader A when it is disabled by Input 5.
To decide if the card is of the Activation type, the Control unit analyses the local card archive, which must therefore be updated.

```
ACT.CARD B ONLY
 1=Yes    3=No
```

By activating this function, the “Activation” or “Activation only” cards are enabled to avoid being ignored by reader B when it is disabled by Input 6.
To decide if the card is of the Activation type, the Control unit analyses the local card archive, which must therefore be updated.

```
VIEW THEN ERR.A
 1=Yes    3=No
```

If this function is activated (it is disabled by default), the Control unit is forced not to immediately display the status of reader A disabled by Input 5. The specified text appears for a few seconds only after a card is read. This function is typically used combined with alarm control units when, for security reasons, one does not want to continuously show the status on the display.

```
VIEW THEN ERR. B
 1=Yes     3=No
```

If this function is activated (it is disabled by default), the Control unit is forced not to immediately display the status of reader B disabled by Input 6. The specified text appears for a few seconds only after a card is read. This function is typically used combined with alarm control units when, for security reasons, one does not want to continuously show the status on the display.

```
NA DEF. REASON
 0-9   ___ (  1)
```

This refers to the default reason associated with a Normal reading (see paragraph) of the card on reader A.
The only case in which this reason is not used, is when you enable the facility to type in a text from 0 to 9 before the card is read. When the card is read, there is only one pending numeric character and there are no enabled “Justifications”.

```
RA DEF. REASON
 0-9   ___ (  2)
```

This refers to the default reason associated with a Reverse reading (see paragraph) of the card on reader A.
The only case in which this reason is not used, is when you enable the facility to type in a text from 0 to 9 before the card is read. When the card is read, there is only one pending numeric character and there are no enabled “Justifications”.

```
NB DEF. REASON
 0-9   ___ (  1)
```

This refers to the default reason associated with a Normal reading (see paragraph) of the card on reader B.
The only case in which this reason is not used, is when you enable the facility to type in a text from 0 to 9 before the card is read. When the card is read, there is only one pending numeric character and there are no enabled “Justifications”.

```
RB DEF. REASON
 0-9   ___ (  2)
```

This refers to the default reason associated with a Reverse reading (see paragraph) of the card on reader B.

The only case in which this reason is not used, is when you enable the facility to type in a text from 0 to 9 before the card is read. When the card is read, there is only one pending numeric character and there are no enabled "Justifications".

```
' ' NA DIRECTION
  1)   2)   3)
```

Three alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. These are the functions:

1) standard behaviour, i.e. the direction associated with the NA reading (i.e. Normal on reader A) is neutral (neither Entry nor Exit).
2) the direction associated with the NA reading (i.e. Normal on reader A) is "**E**" i.e. an Entry.
3) the direction associated with the NA reading (i.e. Normal on reader A) is "**U**" i.e. an Exit.

```
' ' RA DIRECTION
  1)   2)   3)
```

Three alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. These are the functions:

1) standard behaviour, i.e. the direction associated with the RA reading (i.e. Reverse on reader A) is neutral (neither Entry nor Exit).
2) the direction associated with the RA reading (i.e. Reverse on reader A) is "**E**" i.e. an Entry.
3) the direction associated with the RA reading (i.e. Reverse on reader A) is "**U**" i.e. an Exit.

```
' ' NB DIRECTION
  1)   2)   3)
```

Three alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. These are the functions:

1) standard behaviour, i.e. the direction associated with the NB reading (i.e. Normal on reader B) is neutral (neither Entry nor Exit).
2) the direction associated with the NB reading (i.e. Normal on reader B) is "**E**" i.e. an Entry.
3) the direction associated with the NB reading (i.e. Normal on reader B) is "**U**" i.e. an Exit.

```
' ' RB DIRECTION
  1)   2)   3)
```

Three alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. These are the functions:

1) standard behaviour, i.e. the direction associated with the RB reading (i.e. Reverse on reader B) is neutral (neither Entry nor Exit).
2) the direction associated with the RB reading (i.e. Reverse on reader B) is "**E**" i.e. an Entry.
3) the direction associated with the RB reading (i.e. Reverse on reader B) is "**U**" i.e. an Exit.

6.1.4 Menu "1- Panel programming" – "1-4 Card settings"

```
A CARD MIN.LENG.
 1-37  ___ (  1)
```

This refers to a Magnetic-Stripe code. It indicates the minimum permissible length of a code read by reader A: if it is shorter than this length, the card will be ignored.
This applies to both the 'T' and 'N' prefixes.

```
A CARD MAX.LENG.
 1-37  ___ ( 37)
```

This refers to a Magnetic-Stripe code. It indicates the maximum permissible length of a code read by reader A: if it is greater than this length, the card will be ignored.
This applies to both the 'T' and 'N' prefixes.

```
B CARD MIN.LENG.
 1-37  ___ (  1)
```

This refers to a Magnetic-Stripe code. It indicates the minimum permissible length of a code read by reader B: if it is shorter than this length, the card will be ignored.

This applies to both the ‘T’ and ‘N’ prefixes.

```
B CARD MAX.LENG.
 1-37  ___ ( 37)
```

This refers to a Magnetic-Stripe code. It indicates the maximum permissible length of a code read by reader B: if it is greater than this length, the card will be ignored.
This applies to both the ‘T’ and ‘N’ prefixes.

```
A CHAR.SEQ.START
1-37  ___ (  1)
```

This refers to a Magnetic-Stripe code with an ‘N’ prefix. It is used for cutting off the initial part of the code read by reader A. The starting position of the useful code must be specified. If the length of the read code is shorter than the start of the sequence of characters, the card is ignored.

```
A CHAR.SEQ.END
 1-37  ___ ( 37)
```

This refers to a Magnetic-Stripe code with an ‘N’ prefix. It is used for cutting off the final part of the code read by reader A. The end position of the useful code must be specified. If the length of the read code is shorter than the end of the sequence of characters, the card is ignored.

```
B CHAR.SEQ.START
 1-37  ___ (  1)
```

This refers to a Magnetic-Stripe code with an ‘N’ prefix. It is used for cutting off the initial part of the code read by reader B. The starting position of the useful code must be specified. If the length of the read code is shorter than the start of the sequence of characters, the card is ignored.

```
B CHAR.SEQ.END
1-37  ___ ( 37)
```

This refers to a Magnetic-Stripe code with an ‘N’ prefix. It is used for cutting off the final part of the code read by reader B. The end position of the useful code must be specified. If the length of the read code is shorter than the end of the sequence of characters, the card is ignored.

```
A CARD COD.START
 1-37  ___ (  6)
```

This refers to a Magnetic-Stripe code with an ‘T’ prefix. It is used to specify the starting position of the Card Code in the code read by reader A. If the length of the read code is shorter than the sum of A Card Code Start + A Card Code Length, the card is ignored.

```
A CARD COD.LENG.
1-5   ___ (  5)
```

This refers to a Magnetic-Stripe code with an ‘T’ prefix. It is used to specify how many characters on the code read by reader A should be used to generate the Card Code starting from A Card Code Start position. If the length of the read code is shorter than the sum of A Card Code Start + A Card Code Length, the card is ignored.

```
B CARD COD.START
 1-37  ___ (  6)
```

This refers to a Magnetic-Stripe code with an ‘T’ prefix. It is used to specify the starting position of the Card Code in the code read by reader B. If the length of the read code is shorter than the sum of B Card Code Start + B Card Code Length, the card is ignored.

```
B CARD COD.LENG.
1-5   ___ (  5)
```

This refers to a Magnetic-Stripe code with an ‘T’ prefix. It is used to specify how many characters on the code read by reader B should be used to generate the Card Code starting from B Card Code Start position. If the length of the read code is shorter than the sum of B Card Code Start + B Card Code Length, the card is ignored.

```
A INS.CODE START
 1-37  ___ (  1)
```

This refers to a Magnetic-Stripe code with an 'T' prefix. It is used to specify the starting position of the Installation Code in the code read by reader A. If the length of the read code is shorter than the sum of A Installation Code Start + A Installation Code Length, the card is ignored.

```
A INS.CODE LENG.
 1-5   ___ (  5)
```

This refers to a Magnetic-Stripe code with an 'T' prefix. It is used to specify how many characters on the code read by reader A should be used to generate the Installation Code starting from A Installation Code Start position. If the length of the read code is shorter than the sum of A Installation Code Start + A Installation Code Length, the card is ignored.

```
B INS.CODE START
 1-37  ___ (  1)
```

This refers to a Magnetic-Stripe code with an 'T' prefix. It is used to specify the starting position of the Installation Code in the code read by reader B. If the length of the read code is shorter than the sum of B Installation Code Start + B Installation Code Length, the card is ignored.

```
B INS.CODE LENG.
1-5   ___ (  5)
```

This refers to a Magnetic-Stripe code with an 'T' prefix. It is used to specify how many characters on the code read by reader B should be used to generate the Installation Code starting from Installation B Code Start position. If the length of the read code is shorter than the sum of B Installation Code Start + B Installation Code Length, the card is ignored.

6.1.5 Menu "1- Panel programming" – "1-5 Special functions"

```
A DOOR MANGM.
 1=yes     3=No
```

When this function is activated, management of the following is enabled: door with reader A, door status microswitch connected to input 1, and a door-opener push-button connected to input 2. See paragraph for further details.

```
B DOOR MANGM.
 1=Yes    3=No
```

When this function is activated, management of the following is enabled: door with reader B, door status microswitch connected to input 3, and a door-opener push-button connected to input 4. See paragraph for further details.

```
SING.DOOR MANGM.
 1=Yes    3=No
```

When this function is activated, management of the following is enabled: a single door equipped with reader A, reader B, door status microswitch connected to input 3, and a door-opener push-button connected to input 4. See paragraph for further details.
N.B.: if "Single-Door Management" is activated, the **A** and **B** door Management are **automatically disabled.**

```
A DOOR STA.T.OUT
0-255 ___ ( 15)
```

This refers to the timeout period in which input 1 is disabled (A door status microswitch) following the reading of a valid card or pressure of the door-opener push-button (input 2).

```
B DOOR STA.T.OUT
0-255 ___ ( 15)
```

This refers to the timeout period in which input 3 is disabled (B door status microswitch) following the reading of a valid card or pressure of the door-opener push-button (input 4).

```
INST. CODE TEST
 1=Yes    3=No
```

Prefix "T" is significant while the Control unit is in "Stand-alone" mode. In order to validate a card in this situation, the Installation Code read on the card is normally compared with that of the Control unit.
If this function is disabled, the two codes need no longer coincide.

This applies simultaneously to reader A and B.

```
READER A NOT PIN
  1)   2)   3)
```

Three alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. These are the functions:

1) standard behaviour, i.e. reader A is not in PIN or PINSOST mode
2) reader A in PIN mode
3) reader A in PINSOST mode

```
READER B NOT PIN
  1)   2)   3)
```

Three alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. These are the functions:

1) standard behaviour, i.e. reader B is not in PIN or PINSOST mode
2) reader B in PIN mode
3) reader B in PINSOST mode

```
1 LINE F.PINSOST
 1=Yes    3=No
```

If this function is enabled, PINSOST viewing is forced, while typing it on the top line only and with all numeric characters replaced by an asterisk. It is used to simulate the behaviour of the previous panels.

```
PIN CODE TIMEOUT
0-255 ___ ( 16)
```

This refers to the timeout period in seconds within which the reading of a card associated with a PIN must be completed: if the card is presented first, the PIN must be typed within that timeout. If PIN is typed first, the card must be read within that timeout period.

```
MIN. JUSTIF. No.
0 - 6  ___ (  0)
```

Set to 0 if you do not want at least one key to be typed before the card is read.
Set from 1 to 6 if you do want that number of keys to be typed before the card is read.

```
MAX. JUSTIF. No.
0 - 6  ___ (  6)
```

Set from 1 to 6 the maximum number of keys one can type in before the card is read.
The Control unit does not run any checks and, therefore, we advise you to set a value compatible with the minimum length. The two values may coincide, but do not set a value shorter than the minimum length.

```
B JUST. ON A
 1=Yes    3=No
```

If this function is enabled, the system is forced to expect the reader B Justification to be typed on reader A.
This function is used when only one keyboard/display (reader A), two separate card readers are connected to the Control unit.

```
KEY # -> - (A)
 1)  2)  3)  4)
```

Four alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. The function refers to reader A. You can select the character to be shown on the display when you press the "#" key while the "Justifications" are enabled:

1) standard behaviour: i.e. the dash (' – ') is displayed
2) a comma (',') is displayed. Select if the Justification takes on the "Import" value with decimals.
3) the 'gate' sign ('#') is displayed.
4) a separator Space (' ') is displayed.

```
KEY # -> - (B)
 1)  2)  3)  4)
```

Four alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. The function refers to reader A. You can select the character to be shown on the display when you press the "#" key while the "Justifications" are enabled:

1) standard behaviour: i.e. the dash (' – ') is displayed
2) a comma (',') is displayed. Select if the Justification takes on the "Import" value with decimals.
3) the 'gate' sign ('#') is displayed.
4) a separator Space (' ') is displayed.

```
TYPE REAS. 0-9 A
 1=Yes    3=No
```

If this function is enabled, you may type in a numeric key on reader A before the card is read.
This typed in number replaces the corresponding default Reason. A key need not be typed: in this case, the corresponding default Reason is used.
This setting is not operational if "Justifications " are enabled.

```
TYPE REAS. 0-9 B
 1=Yes    3=No
```

If this function is enabled, you may type in a numeric key on reader B before the card is read.
This typed in number replaces the corresponding default Reason. A key need not be typed: in this case, the corresponding default Reason is used.
This setting is not operational if "Justifications " are enabled.

```
TIME-BAND TEST
 1=Yes    3=No
```

This applies only to the card validation process in "Stand-alone" mode. The text is usually enabled, but it can be disabled if you want all cards in this Control unit not to have any time-band limitations even if some cards have associated limitations in the Wincontrol program.
Operates on both reader A and B.

```
A/B FILTER ACTVS
 1= Yes   3=No
```

By enabling this function, the relay activations associated with the activation cards can be filtered according to reader A or reader B. Such activations are usually performed irrespective of reader.
The filter used to differentiate among reader A and B the standard activation for valid cards is used also for this filtering facility.
Operates only in "Stand-alone" mode.

```
DON'T QUEU.E/U A
 1=Yes    3=No
```

By activating this function, even if the direction is 'E' or 'U', it is forced to ' ' (neutral) for communicating a valid card on reader A (both "On-line" and from the historical archive). Therefore, if certain types of reading specify 'E' or 'U' as the direction, they will be locally significant (e.g.: local counts) but as they are not communicated, they will not influence the Wincontrol program.

```
DON'T QUEU.E/U B
 1=Yes    3=No
```

By activating this function, even if the direction is 'E' or 'U', it is forced to ' ' (neutral) for communicating a valid card on reader B (both "On-line" and from the historical archive). Therefore, if certain types of reading specify 'E' or 'U' as the direction, they will be locally significant (e.g.: local counts) but as they are not communicated, they will not influence the Wincontrol program.

```
EXIT ALWAYS OK
 1=Yes    3=No
```

If this function is enabled, with Antipassback active, the exits (readers with 'U' direction) are not subjected to a Presence check. Therefore, the Antipassback operates only at Entry. However, the reading at Exit helps to force the read card into Absent status if it was not absent. This function refers to behaviour in "Stand-alone" mode.

```
LITR.COUNT.DISAB
  1)   2)   3)
```

Three alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. This facility is used to activate/disable the "Litre counter" function.

1) standard behaviour; i.e. the "Litre counter" function disabled.
2) "Litre counter" function enabled; counter incremented only on the leading edge of the IN1-AUX input
3) "Litre counter" function enabled; counter incremented both on the leading edge and the trailing edge of the IN1-AUX input

```
LITRE / PULSE
(   10) _____
```

To set the extent by which the counter should be incremented at each received pulse.
If option 2) is activated (i.e. "increment counter on leading edge only"), the counter is incremented by 10 units whenever the IN1-AUX input is in alarm status; nothing happens when it is restored.
If option 3) is activated (i.e. "increment counter on both leading and trailing edges"), the counter is incremented by 10 units whenever the IN1-AUX input is in alarm status and this also happens when this input is restored.

```
GENER.CARD ARCH.
 1=Yes    3=No
```

To be used only when the system is being set up. If you select 1 (Yes), the mask with the confirmation request appears. If this function is activated, a Card archive is automatically generated, numbered from 1 up to the memory's maximum capacity (e.g.: 1500 or 5000). Some additional archives – such as Time-Bands and Holidays, are also generated.
Attention! the current archives are overwritten. A "Global download" from the Centre is sufficient to restore the situation, overwriting the archives generated by this function.
When the archives have been generated, all the read cards should be valid providing:

- the Card number is between 1 and the maximum figure
- the Card's Installation Code coincides with the one set on the Control unit
- Prefix T must be set and, therefore, it makes no sense working with Passive cards

6.1.6 Menu "1- Panel programming" – "1-6 Input settings"

```
LOGIC INP1 NC
 1=Yes    3=No
```

With this mask you can reverse the function logic of input 1. The input is set as NO by default value(i.e. it should be coupled to a Normally Open contact).
This means that if it is NO:

- if the terminal is free (not connected to GND), the input is considered at rest.
- if the terminal is connected to GND, the input is considered activated or in alarm status.

If the NC (Normally Closed) logic is set, the behaviour is as follows:

- if the terminal is connected to GND, the input is considered at rest.
- if the terminal is free (not connected to GND), the input is considered in alarm status.

```
LOGIC INP2 NC
 1=Yes    3=No
```

See explanations for INP1

```
LOGIC INP3 NC
 1=Yes    3=No
```

See explanations for INP1

```
LOGIC INP3 NC
 1=Yes    3=No
```

See explanations for INP1

```
LOGIC INP4 NC
 1=Yes    3=No
```

See explanations for INP1

```
LOGIC INP5 NC
 1=Yes    3=No
```

See explanations for INP1

```
LOGIC INP6 NC
 1=Yes    3=No
```

See explanations for INP1

```
LOGIC INP7 NC
 1=Yes    3=No
```

See explanations for INP1
Note the following:

- input 7 is dedicated to the Tamper
- the default logic is NC (Normally Closed), i.e. contrary to the other 6 inputs.

```
SAVE ALARM 1 HST
 1=Yes    3=No
```

If this function is activated in "Stand-alone" mode, the **Alarms of input 1** are saved in the events local memory (historical) and are then downloaded to the Wincontrol program as soon as communication is re-established.
As a default setting, Alarm 1 is not saved in order to reserve all the memory for valid cards.

```
SAVE ALARM 2 HST
 1=Yes    3=No
```

See explanations for ALARM 1.

```
SAVE ALARM 3 HST
 1=Yes    3=No
```

See explanations for ALARM 1.

```
SAVE ALARM 4 HST
 1=Yes    3=No
```

See explanations for ALARM 1.

```
SAVE ALARM 5 HST
 1=Yes    3=No
```

See explanations for ALARM 1.

```
SAVE ALARM 6 HST
 1=Yes    3=No
```

See explanations for ALARM 1.

```
SAVE ALARM 7 HST
 1=Yes    3=No
```

See explanations for ALARM 1.

```
SAVE REST. 1 HST
1=Yes     3=No
```

If this function is activated in "Stand-alone" mode, the **Restore operations of input 1** are saved in the events local memory (historical) and are then downloaded to the Wincontrol program as soon as communication is re-established.
As a default setting, Restore 1 is not saved in order to reserve all the memory for valid cards.

```
SAVE REST. 2 HST
 1=Yes    3=No
```

See explanation for RESTORE 1.

```
SAVE REST. 3 HST
 1=Yes    3=No
```

See explanation for RESTORE 1.

```
SAVE REST. 4 HST
 1=Yes    3=No
```
See explanation for RESTORE 1.

```
SAVE REST. 5 HST
 1=Yes    3=No
```
See explanation for RESTORE 1.

```
SAVE REST. 6 HST
 1=Yes    3=No
```
See explanation for RESTORE 1.

```
SAVE REST. 7 HST
 1=Yes    3=No
```
See explanation for RESTORE 1.

```
DELAY ENGAGE IN1
0-255  ___ (  0)
```
The minimum time during which the input in question must be engaged before the Control unit considers it active. This facility is for filtering status variations which are shorter than necessary and is used, in particular, with Loops in Car Parks.

```
DELAY ENGAGE IN2
0-255  ___ (  0)
```
See explanation for IN1

```
DELAY ENGAGE IN3
0-255  ___ (  0)
```
See explanation for IN1

```
DELAY ENGAGE IN4
0-255  ___ (  0)
```
See explanation for IN1

```
DELAY ENGAGE IN5
0-255  ___ (  0)
```
See explanation for IN1

```
DELAY ENGAGE IN6
0-255  ___ (  0)
```
See explanation for IN1

```
DELAY DISENG.IN1
0-255  ___ (  0)
```
The minimum time during which the input in question must be disengaged before the Control unit considers it disabled. This facility is for filtering status variations which are shorter than necessary and is used in particular with Loops in Car Parks.

```
DELAY DISENG.IN2
0-255  ___ (  0)
```
See explanation for IN1

```
DELAY DISENG.IN3
0-255  ___ (  0)
```
See explanation for IN1

```
DELAY DISENG.IN4
```

```
0-255  ___ (  0)
```

See explanation for IN1

```
DELAY DISENG.IN5
0-255  ___ (  0)
```

See explanation for IN1

```
DELAY DISENG.IN6
0-255  ___ (  0)
```

See explanation for IN1

6.1.7 Menu "1- Panel programming" – "1-7 Output settings"

```
TIMEOUT1 (TENTH)
(   10) _____
```

Relay 1 activation time for a valid card (in tenths of a second).
Values from 0 to 65535 are acceptable.
By setting a certain time in the Wincontrol program (e.g.: 200) and the Control unit is set in Seconds (see Tenths/Secs), the set value will be 2000 after the data are downloaded, because the Control unit multiplies 200 x 10 to nevertheless express the time in tenths.
If it were set in Tenths, the received value is left unchanged, i.e. 200.

```
OUT1 A + B (■■)
 1)  2)  3)  4)
```

Four alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. The functions are referred to OUT1 (relay 1):

1) standard behaviour, i.e. the activation is executed whether reading took place on A or on B (■■)
2) activation is executed only if reading was on A (■)
3) activation is executed only if reading was on B (■)
4) there is no activation ( ), as if time were zero.

```
TIMEOUT2 (TENTH)
(   10) _____
```

Relay 2 activation time for a valid card (in tenths of a second).
Values from 0 to 65535 are acceptable.
By setting a certain time in the Wincontrol program (e.g.: 200) and the Control unit is set in Seconds (see Tenths/Secs), the set value will be 2000 after the data are downloaded, because the Control unit multiplies 200 x 10 to nevertheless express the time in tenths.
If it were set in Tenths, the received value would be left unchanged, i.e. 200.

```
OUT2 A + B (■■)
 1)  2)  3)  4)
```

Four alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. The functions are referred to OUT2 (relay 2):

1) standard behaviour, i.e. the activation is executed whether reading took place on A or on B (■■)
2) activation is executed only if reading was on A (■)
3) activation is executed only if reading was on B (■)
4) there is no activation ( ), as if time were zero.

```
TIMEOUT3 (TENTH)
(   10) _____
```

Relay 3 activation time for a valid card (in tenths of a second).
Values from 0 to 65535 are acceptable.
By setting a certain time in the Wincontrol program (e.g.: 200) and the Control unit is set in Seconds (see Tenths/Secs), the set value will be 2000 after the data are downloaded, because the Control unit multiplies 200 x 10 to nevertheless express the time in tenths.
If it were set in Tenths, the received value would be left unchanged, i.e. 200.

```
OUT3 A + B (■■)
1)  2)  3)  4)
```

Four alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. The functions are referred to OUT3 (relay 3):

1) standard behaviour, i.e. the activation is executed whether reading took place on A or on B (■■)
2) activation is executed only if reading was on A (■)
3) activation is executed only if reading was on B (■)
4) there is no activation ( ), as if time were zero.

```
TIMEOUT4 (TENTH)
(   10) _____
```

Relay 4 activation time for a valid card (in tenths of a second).

Values from 0 to 65535 are acceptable.
By setting a certain time in the Wincontrol program (e.g.: 200) and the Control unit is set in Seconds (see Tenths/Secs), the set value will be 2000 after the data are downloaded, because the Control unit multiplies 200 x 10 to nevertheless express the time in tenths.
If it were set in Tenths, the received value would be left unchanged, i.e. 200.

```
OUT4 A + B (■■)
 1)  2)  3)  4)
```

Four alternatives are possible – by activating the numeric key, the corresponding description appears on the top line. The functions are referred to OUT4 (relay 4):

1) standard behaviour, i.e. the activation is executed whether reading took place on A or on B (■■)
2) activation is executed only if reading was on A (■)
3) activation is executed only if reading was on B (■)
4) there is no activation ( ), as if time were zero.

```
OUT FOR KO CARD
1-4   ___ (001)
```

The relay to be activated in case of an incorrect (KO) card, whether read on A or B.
The value can be in the range from 1 to 4.

```
KO CARD TIME
0-255  ___ (  0)
```

The relay activation time in case of an wrong (KO) card, whether read on A or B. The value can be in the range from 0 to 255. Time will be in Tenths or Seconds according to the Tenths/Secs setting.

6.1.8 Menu "1- Panel programming" – "1-8 Serial settings"

If you select this item, you access a level 3 sub-menu.
You can select whether to execute the settings for **COM1** or **COM2**.

```
1-8-1 SETTINGS
      COM 1
```

```
1-8-2 SETTINGS
      COM 2
```

6.1.8.1 Settings for COM1

```
3=POLLING /P
1=MODEM-TERM. /T
```

To modify the protocol mode on the COM1 serial port.
Use Polling (/P) if connected to "Spacenet" in RS485 or connected directly to the PC in RS232.
Use "Modem / Terminal Server" (/T) if a Modem or Terminal Server on LAN Ethernet (TCP/IP) is coupled to the Control unit.

```
BAUD COM1 IF /P
1=← 57.600 3=→
```

To modify Baud Rate of serial port COM1 if the via Polling (/P) protocol is active.

```
BAUD COM1 IF /T
1=← 57.600 3=→
```

To modify Baud Rate of serial port COM1 if the via Modem (/T) protocol is active.

```
WITH CODE (/T)
 1=Yes    3=No
```

If the (/T) communication is enabled – "With code (/T)" = Yes – the Control unit is normally "Shut down";
Communication between the two devices takes place only if the correct "Release Code" is received from the caller.

Activate it only if the communication is via Modem. Do not activate it if via Terminal-Server.

```
CODE 1
0-255  ___ (255)
```

Set “Release code No. 1”. Indicate the 1st of the numbers in item “Guide” – “Information” – “SA remote Code” in the Wincontrol program.

```
CODE 2
0-255  ___ (255)
```

Set “Release code No. 2”. Indicate the 2nd of the numbers in item “Guide” – “Information” – “SA remote Code” in the Wincontrol program

```
CODE 3
0-255  ___ (255)
```

Set “Release code No. 3”. Indicate the 3rd of the numbers in item “Guide” – “Information” – “SA remote Code” in the Wincontrol program

```
CODE 4
0-255  ___ (255)
```

Set “Release code No. 4”. Indicate the 4th of the numbers in item “Guide” – “Information” – “SA remote Code” in the Wincontrol program

```
CODE 5
0-255  ___ (255)
```

Set “Release code No. 5”. Indicate the 5th of the numbers in item “Guide” – “Information” – “SA remote Code” in the Wincontrol program

```
CODE 6
0-255  ___ (255)
```

Set “Release code No. 6”. Indicate the 6th of the numbers in item “Guide” – “Information” – “SA remote Code” in the Wincontrol program

```
T-OUT F.AT SEND.
0-255  ___ (180)
```

Set the character reception absence time which, when it elapses, causes the AT character to be sent on the serial port.

```
DELAY RTS COM1
0-255  ___ ( 64)
```

Set a number in proportion to the delay between activation of RTS (it becomes the transmission command in RS485) and the effective sending of the first character.
Typically, there is no need to change it.

6.1.8.2 Settings for COM2

```
BAUD RATE COM2
1=←  9.600 3=→
```

To modify the Baud rate of serial port COM2 if at least one associated function is active (e.g.: card printing OK).

```
3=8 BIT COM2
1=7 BIT COM2
```

To modify the number of bits used to communicate each character received/transmitted from/to serial port COM2.

```
PARITY NO. COM2
  1)   2)   3)
```

To modify the type of parity used to receive/transmit each character from/to serial port COM2.
This can be “N” (parity No.), “E” (Even), “O” (Odd).

```
3=1 STOP BIT
```

```
1=2 STOP BIT
```
To modify the number of Stop bits used to communicate each character received/transmitted from/to serial port COM2.

```
CARD PRINT OK A
 1=Yes    3=No
```
To enable printing (via serial port COM2 – RS232) of "Valid cards" on reader A.

```
CARD PRINT OK B
 1=Yes    3=No
```
To enable printing (via serial port COM2 – RS232) of "Valid cards" on reader B.

```
CHARACTER OUTPUT
 1=Yes    3=No
```
To enable the function allowing connection of an Indicator board to serial port COM2 – RS232.

```
READ CARDS COM2
 1=Yes    3=No
```
To enable the function allowing connection of a Barcode Card Reader (or reader with an RS232 output) to serial port COM2 – RS232

6.1.9 Menu "1- Panel programming" – "1-9 Car Park settings"

```
RESET PRES.CARDS
 1=Yes     3=No
```
This is used for aligning the local "**Count**" (if activated). We do not recommend using it if you are connected on-line to the Wincontrol program.
It operates directly in the RAM on the **local** Card archive. It forces all cards to "Absent", and the "Present Cards" counter equals zero (NB.: the "Total present" becomes equal to the "No. of fraud events" if enabled).
It asks for a **"Yes/No" confirmation** before resetting.

```
TOTAL PRESENT
(   10) _____
```
This is used for aligning the local "**Count**": use it only for "**Car Park**" with "**Fraud events"** activated.
Operates directly in the RAM.
Before displaying the above mask, the system indicates the counts situation for 5 seconds: the **first** line shows the "**Total present**". The **second** line shows the "**Present cards**" and "**No. of fraud events**" in that order.
One has to type in the number of vehicle actually counted inside the Car Park. The system calculates the "No. of Fraud events" according to the "Present cards", displaying the result of the count for 5 seconds.
It then asks for a **"Yes/No" confirmation** before updating the "Total present" counter.

```
A LANE NORMAL
  1)   2)   3)
```
This is used only for a "**Car Park**". Operates directly in the RAM.
It is able to perform the same commands which can be given from the Wincontrol program using the "Push-button panel":

- **A** lane normal ("Normal operation")
- **A** lane always open ("Permanently open")
- **A** lane always closed ("Permanently closed")

```
B LANE NORMAL
  1)   2)   3)
```
This is used only for a "**Car Park**". Operates directly in the RAM.
It is able to perform the same commands which can be given from the Wincontrol program using the "Push-button panel":

- **B** lane normal ("Normal operation")
- **B** lane always open ("Permanently open")
- **B** lane always closed ("Permanently closed")

```
A TRANSITS MANGM
 1=Yes    3=No
```
Enables or disables management of “Transits” on reader A. **Attention:** it enables communication of a second event, after the card is read, which may be relevant to counts, antipassback, etc. **Consult the relevant paragraph**

```
B TRANSITS MANGM
 1=Yes    3=No
```
Enables or disables management of “Transits” on reader B. **Attention**: it enables communication of a second event, after the card is read, which may be relevant to counts, antipassback, etc. **Consult the relevant paragraph**

```
A CAR PARK
 1=Yes    3=No
```
Enables or disable management of the “Car Park” on reader A. If disabled, that means that A is a pedestrian gateway.

```
B CAR PARK
 1=Yes    3=No
```
Enables or disable management of the “Car Park” on reader B. If disabled, that means that B is a pedestrian gateway.

```
A PARKING MODE
 1-4   ___ (  1)
```
Used for setting the “Reader A Parking mode”. A loops installed situation corresponds to each number: 1 = no loop; 2 = “Presence”; 3 = “Transit” ; 4 = “Presence” + “Transit”.
Also used for a “Pedestrian gateway” with “Transits” enabled (see paragraph)

```
B PARKING MODE
 1-4   ___ (  1)
```
Used for setting the “Reader B Parking mode”. A loops installed situation corresponds to each number: 1 = no loop; 2 = “Presence”; 3 = “Transit”; 4 = “Presence” + “Transit”.
Also used for a “Pedestrian gateway” with “Transits” enabled (see paragraph)

```
SINGLE BEAM MODE
 1=Yes    3=No
```
This is used for setting “Single beam” management, i.e. lane A and lane B coincide and are opposed to each other.
It makes sense only if lane A and lane B are in “Car Park” = Yes. See paragraph.

```
A TRANS. TIMEOUT
0-255  ___ ( 10)
```
Set the Transit Timeout period on reader A. Can be used if the “Transits” are active (also for a “pedestrian gateway”). Can also be used for a “vehicle gateway” and disabled “Transits”, because it determines the beam re-closing time.

```
B TRANS. TIMEOUT
0-255  ___ ( 10)
```
Set the Transit Timeout period on reader B. Can be used if the “Transits” are active (also for a “pedestrian gateway”). Can also be used for a “vehicle gateway” and disabled “Transits”, because it determines the beam re-closing time.

```
B BEAM PARK.LOG.
 1=Yes    3=No
```
For enabling activation of A beam using two relays: one for raising the beam, and the other for lowering it.
Can be used only if the lane is in “Parking” mode.

```
B BEAM PARK.LOG.
 1=Yes    3=No
```
For enabling activation of B beam using two relays: one for raising the beam, and the other for lowering it.
Can be used only if the lane is in “Parking” mode.

```
A CL. EVERY MIN.
 1=Yes    3=No
```
Enables activation of the “Close pulse” for A beam for every full-status minute in order to repeat closure.

Can be used only if in “Parking” mode and with A beam in “Parking Logic”.

```
B CL. EVERY MIN.
 1=Yes    3=No
```

Enables activation of the “Close pulse” for B beam for every full-status minute in order to repeat closure.
Can be used only if in “Parking” mode and with A beam in “Parking Logic”.

```
A FRAUD EVENTS
 1=Yes    3=No
```

For enabling “Fraud events” management on lane A. Can be used only if the lane is in “Parking” mode.

```
B FRAUD EVENTS
 1=Yes    3=No
```

For enabling “Fraud events” management on lane B. Can be used only if the lane is in “Parking” mode.

```
A LANE ENTRY
  1)   2)   3)
```

Use it only for “**Car Park**” with “**Fraud events”** enabled. When a “Fraud event” is detected, it is communicated (or managed locally if in Stand-alone mode) with this information associated. These are the options:

1. A lane Entry (“**E**”)
2. A lane Exit (“**U**”)
3. A lane Neutral (i.e. it must not influence counts)

```
B LANE EXIT
  1)   2)   3)
```

Use it only for “**Car Park**” with “**Fraud events”** enabled. When a “Fraud event” is detected, it is communicated (or managed locally if in Stand-alone mode) with this information associated. These are the options:

1. B lane Entry (“**E**”)
2. B lane Exit (“**U**”)
3. B lane neutral (i.e. it must not influence the counts)

```
PRIV.CARDS MANGM
 1=Yes    3=No
```

For enabling or not enabling the “Privileged cards management”. With this function, the cards can be sub-divided into groups (according to the Presence Detection Code) and each group behaves in a certain way when the car park is in excess mode. For further information, see the relevant paragraph.

6.1.10 Menu “2 – EEPROM direct programming” – “2-1 EEPROM location writing”

```
LOCAT.: ___
VALUE:___ (255)
```

In the top line, indicate a number from 001 to 2048 + Enter - this is the location to read/write.
The current value in brackets appears on the bottom line, and the cursor flashes signalling that you can type in a number from 0 to 255 – this is the decimal value to write (255=ff hex).
If you press only the Enter key without typing in any value, the current value is **confirmed** (in practice, this serves to read the status of a location.) Read and typed in values are decimal.

6.1.11 Menu “2 – EEPROM direct programming” – “2-2 EEPROM clean-up”

Before executing the command, the Control unit shows the following mask to avoid unwanted cancellations.
The cursor flashes over “No” and, if you want to execute, shift it to “Yes”.

```
ARE YOU SURE?
 1=Yes    3=No
```

The command executes a new initialisation of the values in EEPROM not including the following which remain unchanged:

- Panel No.

- installation code
- language
- maximum No. of cards

It is similar to the next command apart from the fact that the latter does not change the value of those in the list.

6.1.12 Menu "2 – EEPROM direct programming" – "2-3 EEPROM clean-up and default ":

Before executing the command, the Control unit shows the following mask to avoid unwanted cancellations.
The cursor flashes over "No" and, if you want to execute, shift it to "Yes".

```
ARE YOU SURE?
 1=Yes    3=No
```

The commands executes a new initialisation of all the values in EEPROM, loading default values:
Note that the panel is re-numbered as No. 255, set as /P, Italian language, maximum No. of cards set to equal the default number and Installation Code set on 65535.

6.1.13 Menu "3- Language" – "3-1 Italiano"

If this item is selected, from that moment on all texts will be in Italian (programming included).
After the selection, you stay in programming mode, and the previous menu level is presented. The texts are already in Italian.
If a black spot in brackets appears on the top line, this means that the displayed language is the current one.

6.1.14 Menu "3- Language" – "3-2 English"

If this item is selected, from that moment on all texts will be in English (programming included).
After the selection, you stay in programming mode, and the previous menu level is presented. The texts are already in English.
If a black spot in brackets appears on the top line, this means that the displayed language is the current one.

6.1.15 Menu "3- Language" – "3-3 Español"

If this item is selected, from that moment on all texts will be in Spanish (programming included).
After the selection, you stay in programming mode, and the previous menu level is presented. The texts are already in Spanish.
If a black spot in brackets appears on the top line, this means that the displayed language is the current one.

6.2 Programming without keyboard

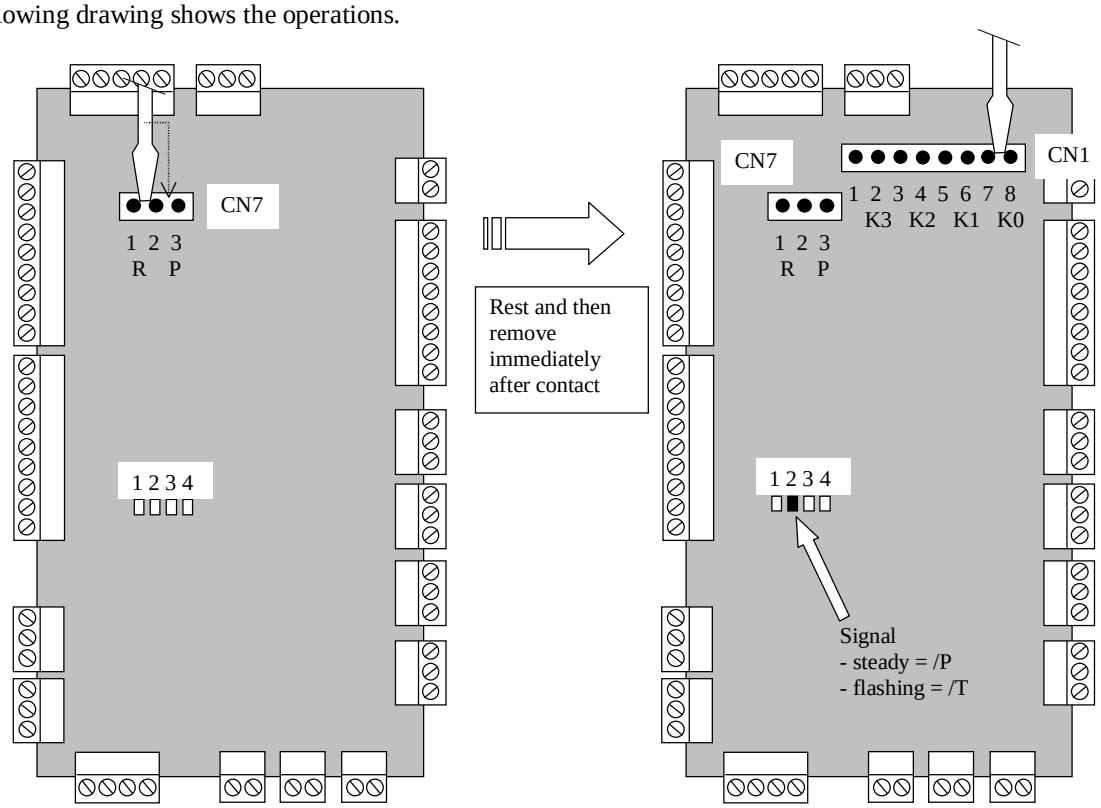
Only a few programming operations are possible if the keyboard/display connected to the Control unit does not communicate with the PC. Only two operations are possible:

1. changeover the mode from /**P** to /**T** and vice versa
2. **restore** the **default** settings

The possibilities described below are important because all you have to do is to operate on the board with a simple metal object (e.g.: screwdriver, key) to make the settings.
In any case, you have to access the programming mode using the CN7 3-pole connector as described at the beginning of the keyboard programming. Next, you have to work on the CN1 8-pole connector, making short-circuits between the pins to executed commands, as described in detail below.
N.B.: as some signals are produced by using the Buzzer fitted on the board, make sure it is ON.

6.2.1 Switching between /P and /T

First of all, enter the programming mode.
Remember that LED2 signals the current status (if lighted steadily it is /P, if flashing, it is /T).
Next, adjust Pin 7 and 8 of Connector CN1 to switch the current setting.
Create a short-circuit between these pins until you hear the Buzzer, then release the contact and check LED2 to verify if the switch was made.
This operation can be carried out several times.
Reset to return to the "operational mode".
The following drawing shows the operations.



Phase 1) and 2):
Reset + Programming

Phase 3):
Switch from /P to /T or vice versa

6.2.2 Restore the default settings

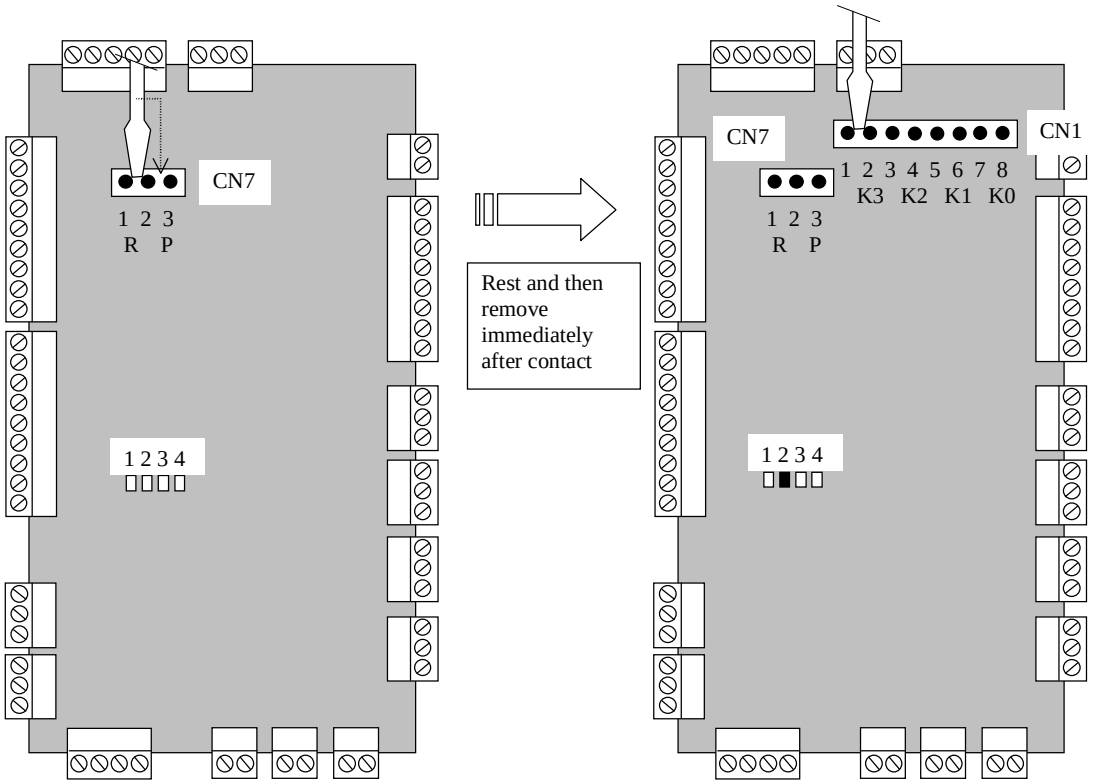
First of all, enter the programming mode.
Next adjust Pin 1 and 2 of connector CN1 to activate the restoring mechanism
Procedure :

1. create a short-circuit between pins 1 and 2 until you hear the Buzzer
2. release until you hear the Buzzer again
3. create a new short-circuit between pins 1 and 2 until you hear the Buzzer

At this point, release the contact and check the Reset LED to see if restored.
If the board is reset, turn off LED 2 informing you if you are in programming mode.

The effected **restore operation** is exactly the same as the one which would occur by selecting the "2-3 EEPROM CLEAN-UP AND DEFAULT" item of the keyboard programming menu.

The following drawing shows the operations.



Phase 1) and 2):
Reset + Programming

Phase 3):
Activate the contact as in the text

7 “Via Modem” Control unit

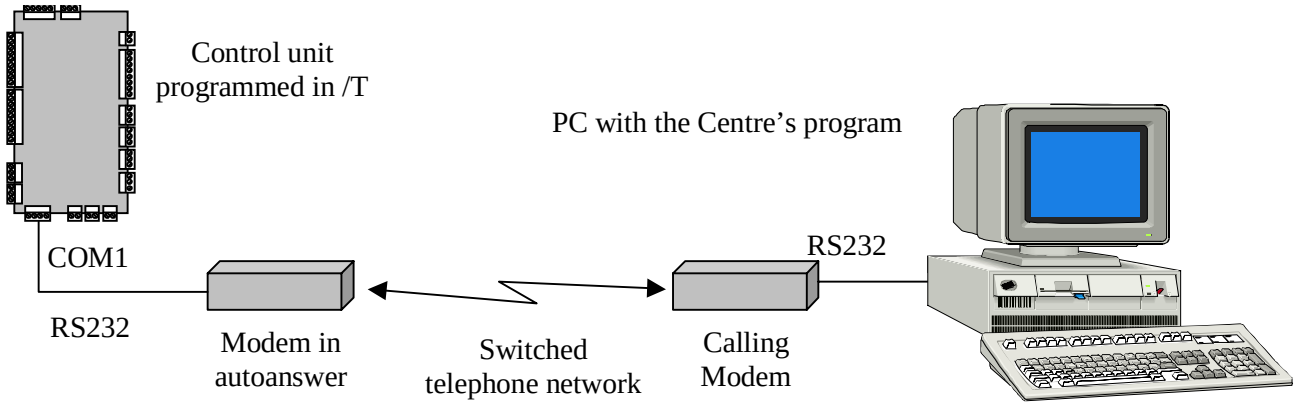
The **Control unit** must be configured in “/**T**” (communication via Modem/Terminal Server).
The Wincontrol program can communicate with the Control units in three different modes:

1. via Polling
2. via Modem (only if set as “On-line” “/T”)
3. via Terminal Server on LAN (only if set as “On-line” “/R”)

One mode excludes the other.
The Polling mode is the classic “Spacenet” in RS485 (or in RS232 if the connection is direct)
The via Modem mode is employed when the distances rule out laying a cable for RS485 and make it necessary to use a switched telephone line if available.
The following are at least necessary:

- a Modem (external or internal) in the PC where the Wincontrol program is installed
- an external modem for each Control unit
- a connection cable to be made as described in the relevant paragraph.

The modem must be Hayes-compatible. It should be approved by the postal service in your country.
The Control unit must be programmed for “via Modem” (/T) use.
After being set in this way, the Control unit can no longer communicate via Polling with the Wincontrol program.
The Control unit connected “via Modem” operates in “Stand-Alone” mode and not “On-Line” as in the case of the RS485 connection.

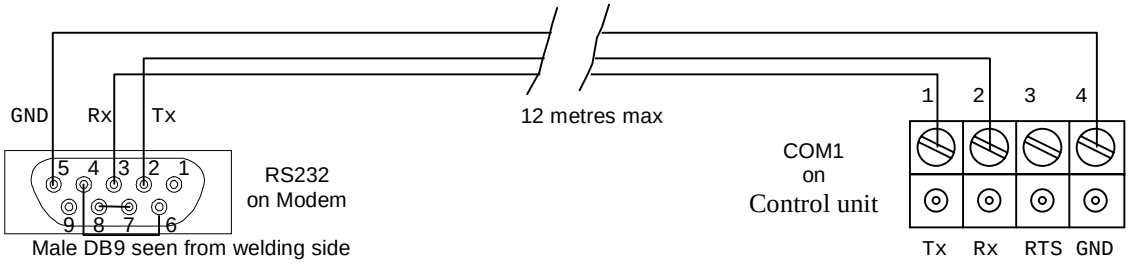


7.1 Connections to the Control unit

Place the Modem in a protected position:

- near to a telephone socket
- near a power socket

Make a cable as described and connect it to the COM1 port of the Control unit (removable terminal board) and to the Modem (DB9 or DB25).
Connection cable between Modem and Control unit if the Modem has a female **DB9** connector.

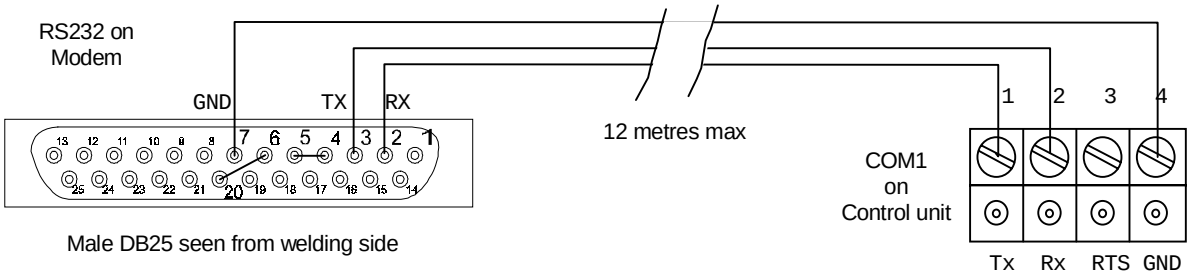


N.B.: Do not forget the two **jumpers** between :

- Pin 4 and Pin 6
- Pin 7 and Pin 8

The declared distance is the official distance of the RS232 standard valid up to Baud rate of 115,200.

Connection cable between Modem and Control unit if the Modem has a female **DB25** connector.



N.B.: Do not forget the two **jumpers** between :

- Pin 4 and Pin 5
- Pin 6 and Pin 20

The declared distance is the official distance of the RS232 standard valid up to Baud rate of 115,200.

7.2 Connection to PC

The PC on which the Wincontrol program is installed must have a free serial port by other functions and dedicated to the calling function only of panels marked "ON-LINE"/"T".
The following number of serial ports are required:

- 3 serial ports (e.g.: COM1, COM2, COM3) if present
  - mouse on RS232
  - Polling port ("Spacenet" in RS485) for communicating with one or more "via Polling" Control units.
  - port for Transmission from connector to Modem
- 2 serial ports (e.g.: COM1, COM2) if present
  - mouse on RS232
  - port for Transmission from connector to Modem
- 2 serial ports (e.g.: COM1, COM2) if present
  - Polling port ("Spacenet" in RS485) for communicating with one or more "via Polling" Control units.
  - port for Transmission from connector to Modem
  - mouse on port PS2
- 1 serial port (e.g.: COM1) if present
  - port for Transmission from connector to Modem
  - mouse on port PS2

Connect the serial cable supplied with the Modem between:

- the Modem itself
- the free serial port of the PC which will become the "Transmission port".

It is at the user's complete discretion to decide which port to use for Polling and which for Transmission (for the Modem).
Remember that the operating system does not allow two programs using the same serial port to be executed simultaneously.

7.3 Settings on the Wincontrol program

- Select menu item "Initialisations" – "Serial data"
- Confirm the items of the first mask (or press ESC) referring to the "Polling Port" and make the second mask – described as "Transmission Port" - appear.
- Check if the settings of the "Data transmission/reception" port are the same as those of the Control unit and are compatible with the supplied telephone channel and the Modem being used (the default value is 19,200 Baud and can be a good compromise between transfer speed, the line's capability to operate at certain speeds and the Modem's capability to operate at certain speeds (consult the manual). Do not, in any case, use 115,200
- Check if the settings of the "Data transmission/reception" port are "N, 8, 1, Modem Yes".

Remember that changing over from 19,200 to 57,600 Baud rate does not mean the communication lasts one third of the time: there is an improvement but not of the expected proportions.
The initialisation string could be:

```
~~ATE0 V1 B0 X3 L3 S0=1 S6=2^M~~
```

Note: the SØ=1 command enables autoanswer and, in particular, the Modem connects to the line at the first ring.

There is strictly no need to activate autoanswer (SO=1) but, in some cases, if the Modem is used for other services, it should be enabled.

Select menu item “Archives” – “Panels” and set all panels equipped with a Modem as follows:
- set the type as “**ON-LINE**”/“**T**”
- select item “Telephone No.”
- set the “Telephone No.” which must be dialled by the Wincontrol Modem in order to reach the peripheral Modem
- If necessary, set the “Automatic call hours”
- Set the number of attempts
- Set 5 maximum calls time for each day of the week

7.4 Settings on the Control unit

Follow this procedure:
1. Open the “Guide” – “Information” mask in the Wincontrol program. Take a note of the 6 numbers: **Remote SAs code**
2. Access the Control unit’s keyboard programming and select menu item “Serial settings”.
3. When “WITH CODE” is prompted, answer “Yes” (key ‘1’)
4. When “CODE 1” is prompted, type in the first annotated number **(Remote SAs Code)**
5. When “CODE 2” is prompted, type in the second annotated number **(Remote SAs Code)**
6. When “CODE 3” is prompted, type in the third annotated number **(Remote SAs Code)**
7. When “CODE 4” is prompted, type in the fourth annotated number **(Remote SAs Code)**
8. When “CODE 5” is prompted, type in the fifth annotated number **(Remote SAs Code)**
9. When “CODE 6” is prompted, type in the sixth annotated number **(Remote SAs Code)**
10. When “BAUD IN /T” is prompted, set the same value as set in the Wincontrol program.

Note that the device is already programmed as **”STAND-ALONE ONLY”.** Do not modify this and do not input “ON-LINE ONLY”:
However, other programming operations linked to the system can be effected (e.g.: T/N prefix, Wiegand/MS, Presence Detection mode, etc.)

7.5 Settings on Modem

The Modem should be initialised with parameters making it operational at each power-up.
The basic functions to be activated are (the commands to be sent if using a communication program are shown in brackets):
- disable local eco (“ATEØ”)
- enable autoanswer (“ATSØ=1” if it connects the line at the first ring).
- give the command which makes the modifications permanent (“AT&W”)

To facilitate those not having a PC for sending these commands, we advise doing so directly from the Control unit.
Always proceed as follows:
- Switch off both Modem and Control unit
- connect the cable between Modem and Control unit
- power up the Modem first
- then power up the Control unit
- on the Control unit, briefly connect to earth the IN1-AUX input of the auxiliary connector (the 6+6 pole connector near the two transistors). The short-circuit should be made between Pin 10 and 12 which are adjoining. This entails sending the “**ATEØSØ=1&W**” command on serial port COM1 of the Modem as previously described.

N.B.: if using an old Modem, we advise you to use the manufacturer’s default parameters as the internal parameters. To do this, you need a PC which gives the following command by emulating the terminal.

AT&F

7.6 The system’s functionality

The archives must be prepared as in the classic case of terminals connected to “Spacenet”:
First of all, the peripheral unit of the Wincontrol program must be programmed for the first time.
A manual call must therefore be made. Procedure :

- select menu item “Communication” – “Telephone link with Panels”.
- select the Panel number
- put a tick mark in the “Transmit archives” box (☑)
- confirm the call (“OK” push-button)

At this point, a call is made containing all the information obtained from the archives that initialise the remote Panel.

there are two types of calls:
1. the call that simply transfers the data of transits stored by the Control unit to the Wincontrol program.
2. the call that first transfers the transits data stored by the Control unit and then downloads the archives from the Wincontrol program to the Control unit

The following calls can be activated in the Wincontrol program:
- **Automatic** calls: these can be of both types: this depends on the tick mark in the “Transmit archives” box (☑) in the “Automatic call” mask in the “Panel archive”.
- **Manual** calls: these can be of both types: this depends on the tick mark in the “Transmit archives” box (☑) in the mask “Communications” - “Telephone link with Panels”.

Clearly the calls transmitting the archives last longer and, therefore, we advise you to activate them only after making modifications to the archives or when you think that a certain remote Panel does not have the archives aligned with those of the Wincontrol program.

Attention:
- Card readings can be performed while archives are being downloaded from the PC to the Control unit. However, one cannot entirely rule out the possibility of some faults linked to the analysis of an archive while it is being overwritten by the new one. Consequently, we advise you to make the calls - especially automatic calls – at a time of the day when one can assume that card readings are infrequent.
- Readings can be effected while cards are being normally downloaded from the Control unit to the PC – i.e. an automatic call or a manual call without the tick mark on the “Transmit archives” box (☐).
- all parameters can be programmed from the Wincontrol program (see relevant manual), even in the case of readers without a keyboard:
  operate in “Telephone link with panels” activating “Enable Panel configuration”.
- The panel in question cannot be used for a direct **Autoreading**. In practice, you can make use of a collateral function in order to get around this problem. This is not a true autoreading, but, in extreme cases, it can be used to acquire the character Sequence of one or more cards. Contact the supplier for further details. Remember that an Autoreading is possible only if the panel is in “N” Prefix.

8 Control unit “Via LAN (TCP/IP)”

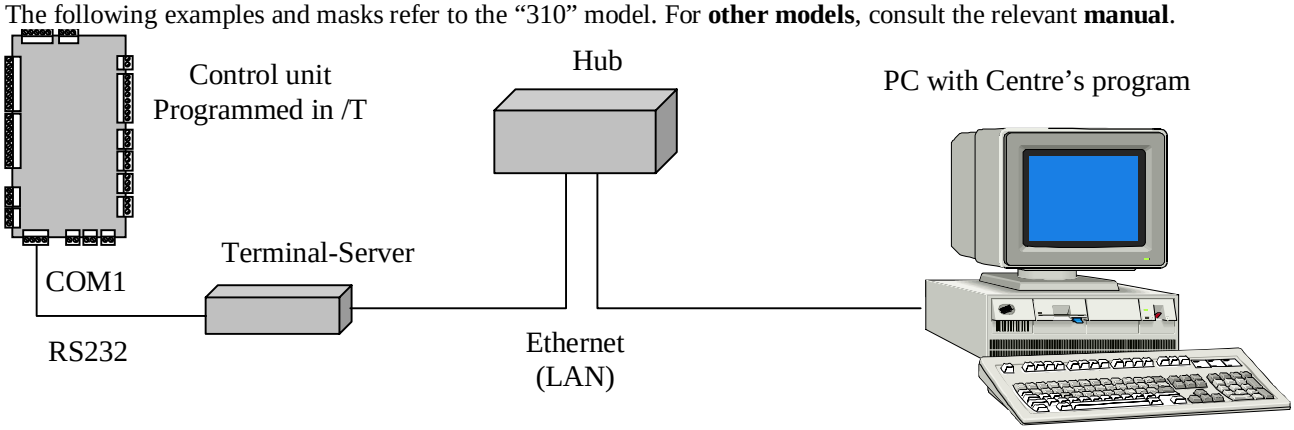
The **Control unit** must be configured in “/**T**” (communication via Modem/Terminal Server).
The Wincontrol program can communicate with the Control units in three different modes:
1. via **Polling**
2. via **Modem** (only if set as “On-line” “/T”)
3. via **Terminal Server** (also known as “**Ethernet box**”) on **LAN TCP/IP** (only if set as “On-line” “/R” in the Panel Archive of the Wincontrol program).

One mode excludes the other.
One makes use of the via network mode (TCP/IP) when it is not convenient to create a “Spacenet” in RS485 and/or a company LAN is already installed.
The Control unit connected via the TCP/IP network operates in “Stand-Alone” mode and not “On-Line” as in the case of the RS485 connection.
This means that the card validity decisions are made by the Control unit and the following limits apply as described in the specific paragraph: maximum number of cards, maximum number of events, etc.
However, the clock-ins are downloaded in real time (within 10 to 15 seconds).

The following are at least necessary:
- A PC with the Wincontrol program installed on it, connected to a LAN with the TCP/IP protocol activated (See Settings – Control panel- Network)
- An adapter of one of the following types: “Ethernet-serial” type “310 Ethernet Box” or “350 Ethernet Box” by Cipher-Lab. This device should be connected, at one end, to the LAN and, at the other, via a serial RS232 to the Control unit.
- A control unit configured as “/T”.

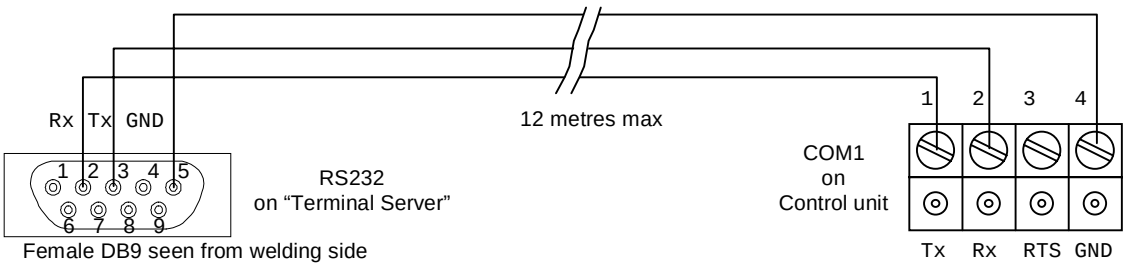
- A network infra-structure (Wiring, Hubs etc.).
- **Autoreading** is possible by interposing a Terminal Server, and operating in exactly the same way as in the case of the Polling panel. Remember that an Autoreading is possible only if the Control unit is in "N" Prefix.

The following examples and masks refer to the "310" model. For **other models**, consult the relevant **manual**.



8.1 Connections between Control unit and Terminal Server

Connection cable between Modem and Control unit if the Modem has a female **DB9** connector



8.2 Description of system

On request, the Control unit can be supplied already connected to the Ethernet-serial adapter (Terminal Server). As a result, all that is needed is to connect the equipment to the network by means of an RJ45 plug (10 Base T) or by a BNC connector (10 Base 2).
The connection should be made to the Control unit's COM1 serial port (the one habitually used for polling).
The Ethernet-serial adapter is supplied by the manufacturer with the IP address as the default setting. The address (e.g.: 192.168.1.1) can be re-programmed to assign another address chosen by the administrator of the company LAN. The configuration of the Ethernet-serial adapter can be modified by connecting to TELNET via LAN on the 8000 port or by an RS232 serial connection. In the latter case, it must be disconnected from the Control unit and connected via RS232 to a PC containing an executed communication program such as "Hyper Terminal".
For further details, consult the manual of the Ethernet-serial adapter.

8.3 "Terminal Server" settings

To configure via serial line, use the accessory program "**TS-CONFIG**" supplied on a floppy disk by the producer of the multi-function control unit and follow the instructions supplied by the program.

To configure via LAN, retrieve the "Execute" command from the "Start" menu and insert "TELNET". When the TELNET program is activated, select "Connect – Remote system…". Input the adapter's address in the "Host Name" box and input the value 8000 in the Port box (for adapters other than Cipher-lab's "310 Ethernet Box", the port value may differ). When you have established connection with the adapter, you can access the different configuration screens, following the instructions shown on the display. In addition, check if the adapter serial communication parameters are identical to those programmed on the Control unit, which has the following default setting:
Flow Control: None, BaudRate: 19200, Parity Bits: None, Data Bits: 8 Bits, Stop Bits: 1 Bit.

Another important parameter in the “Additional Setup Screen” is “Timer-Range” which should be set to 1 (the timer range indicates time in milliseconds and, therefore, the data will be stored in the buffer before they are sent: minimise this value).

The following is a correct example of configuration shown on TELNET, selecting point “5 Display Settings”.

```
 Local Address: 192.000.000.239                Serial NO: 00:C0:02:F4:96:19
 Remote Address: (Invalid)
 Gateway Address: (Invalid)
 Subnet Mask: 255.000.000.000
 Name of Contact Person: Supervisor
 Device Name: SSF49619
 Physical Location: Head Office

 Entry  Manager_IpAddr
 *****  ***************
   1.   000.000.000.000
   2.   000.000.000.000
   3.   000.000.000.000
   4.   000.000.000.000

Current Setting of Serial Port:
 Flow Control: None
 BaudRate: 19200
 Parity Bits: None
 Data Bits: 8 Bits
 Stop Bits: 1 Bit


Current Setting of Buffer:
 Timer: 1 ms
 Block Size: 4096 Bytes
 High Water: 80%
 Low Water: 20%
 Line Terminator Character: 0 (Dec)
 Transmit on LT Char: OFF
 Transmit Mode: NO Filter
 Client Mode: No Echo
 DTR Enable: OFF

Current Configuration of Serial Server:
 Switches Status : SW 1 (ON), SW 2 (ON)  (Server Mode)
 Current Connection Host IP Address: Null
```

If you do not succeed in connecting to the Ethernet-serial adapter, check the following points:

- Is the adapter correctly installed, are the LAN connections right, and is power supplied?
- Are the IP address and the network Mask of your PC correctly set? (see Control panel – Network and properties of the TCP/IP protocol).
- If your PC is not using an IP address with a range compatible with the adapter’s IP address and network Mask, you should add an item in the “arp” table, by typing in the following command when prompted: `arp –s  ip_address harware_address`

For further details, see the “Telnet Configuration” chapter of the adapter manual.

8.4 Settings on the Wincontrol program

Select menu item “Archives” – “Panels” and set all panels connected in LAN as follows:

- set the type as “ON-LINE”/“R”
- press the “IP Address” push-button
- insert the IP address associated with that panel in the mask, and insert the value 3000 in the “port” box (for adapters other than Cipher-Lab’s “310 Ethernet Box”, the port number may differ).

The effective connection of the panels to the network can be directly checked on the monitoring screen or by using the command “Communication” – “Panel Polling control”. The enabled badges must be sent with the command “Communication” – “Download arichives to Panels (DA)”. The Control unit’s specific configurations cannot be made from the Wincontrol program (e.g.: with the command “Programming On-line Panels”), but must be programmed by using the reader’s keyboard directly.

9 Appendix

9.1 News about the Firmware version

The Control unit's program is on Flash: this is a non-volatile memory which can be rewritten hundreds of times.
Every EPROM has:

- the version (e.g.: Ver. 01.00.03)
- the option (OP00, etc.) Only the basic version is available at the moment, i.e. the OP00.

9.1.1 Display of current version

The display can show, for a few seconds, a screen containing some important information.
How to show the screen:

- at power-up (for a few seconds)
- after every Reset (for a few seconds)
- by pressing the key (#)

❖ What is shown on the display's **top** line:
  - the panel's address: E.g.: "N. 001"
  - two letters identifying the type of communication: '/P' if via Polling; '/T' if via Telephone (Modem) or LAN
  - the type of connected reader (A or B)

❖ What is shown on the display's **bottom** line:
  - the version E.g.: "Ver. 01.02.00"
  - the option. E.g.: "00"

The EPROM version consists of three digits

- the two on the left change in case of important updates (e.g.: from 01.00.00 to 02.00.00)
  N.B.: in the case of "firmware update", if at least one of these numbers changes, an EPROM **clean-up** will follow
- the one on the right changes in the event of small program variations (e.g.: from 01.02.00 to 01.02.01)
  N.B.: in the event of "firmware update", if this number changes, **nothing** happens to the EEPROM

9.1.2 Display identifying if reader A or B

The following two techniques are available to know if a reader is 'A' or 'B':
1) If the reader is equipped with a keyboard/display:

- press the "#" key

The panel No., the EPROM version, and letter **A** or **B** identifying the reader will appear on the **display**.

2) If the reader is equipped with a buzzer but without a keyboard/display:

- cut power to the Control unit, wait for 10 seconds and restore power.
  - ➢ if it is reader **A**, the buzzer is briefly sounded **once**
  - ➢ if it is reader **B**, the buzzer is briefly sounded **twice**

9.1.3 Signals on the display

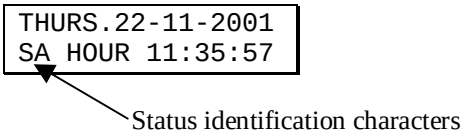
During normal operation, the display shows information regarding the operational status of the Control unit.
In particular, we are referring to the two characters shown (or not shown) low down on the left of the display.
The following tables summarise the situation.

If Control unit set in **/P**

Mode	Forcing	Display
Polling enabled	Normal	nil
No Polling	Normal	**SA**
Polling enabled	Always Stand-alone	**SD**
No Polling	Always Stand-alone	**SA**

If Control unit set in **/T**

Mode	Display
Normal	**SC**
Always Stand-alone	**SC**



9.1.4 LED signalling

There are 4 LEDs on the Control unit for different signalling purposes.
Only **LED 1** and **LED 2** are currently in use (see general table).
The following table describes the associated signals.

Led 1	Led 2	Associated signal
● (Off)	● (Off)	Operational, /**P**, On-line
❄ (lighted)	● (Off)	Operational, /**P**, Stand-alone (might be "Always St.-alone")
❄●❄●❄●❄●❄●❄●❄	● (Off)	Operational, /**P**, "Always Stand-alone"
❄❄●●●❄❄❄●●●❄❄❄	● (Off)	Operational, /**T**
● (Off)	❄ (lighted)	Programming, /**P**
● (Off)	❄●❄●❄●❄●❄●❄●❄	Programming, /**P**, "Always St.-alone"
● (Off)	❄❄❄●●●❄❄❄●●●❄❄❄	Programming, /**T**

Meaning:
❄❄❄●●●❄❄❄●●●❄❄❄ **slow** flashing (lights every second)
❄●❄●❄●❄●❄●❄●❄ **rapid** flashing (lights every tenths)
Note the following:

- **LED 1** and **LED 2** are never lighted simultaneously
- If **LED 2** is lighted, the Control unit is in **Programming** mode

Remember that LED 2 can be used in an alternative way:

- when operating normally in Polling mode, **LED 2** is normally **OFF** but lights up **briefly** when the Control unit is communicating with the PC (e.g.: a card, an alarm, the Firmware version, etc.).

9.1.5 Firmware updating

On the Control unit, the Firmware (i.e. the application program) is stored in a Flash memory which can be rewritten many times. This is a non volatile memory but cannot be removed as it is directly welded onto the printed circuit.
However the Firmware can be updated by downloading the new Firmware version via serial cable from a PC.
Local downloading on the Auxiliary **COM2** port is possible on all versions.
As from Ver. 01.01.00, updating on the **COM1** port from a **remote computer** is also possible (e.g.: the PC where the Wincontrol program is installed). This means that the network structure normally enabling communication between the PC and the Control unit can be used for updating the firmware. If the Control unit is connected in Polling mode (/P) in RS485, this channel can be used without the need to physically access the Control unit.
This also applies for LAN or Modem (Control unit in /T) connections.
The Firmware can be downloaded from:

- the Wincontrol program providing it is sufficiently recent (check if there is a button with the Eprom symbol and a red arrow in the "Communication" – "Panel configuration" mask).
- "File2com" program

The following are needed for the updating procedure:

- a PC with a Windows operating system (95, 98, 2000, NT) and a free serial port.
- the program suitable for downloading data (see above)
- the new file to be sent to the Control unit, containing the new Firmware (typically a ***.bin file).
- a three-wire connection cable described further below (but only in the case of direct downloading on COM2).

"Linear" downloading

The firmware is downloaded in one operation. The Control unit must have an adequate RAM for temporarily storing the firmware while waiting for it to be copied on Flash.
This can be done by just using the Control unit's COM2 port.

N.B.: "linear" downloading is more reliable than "block" downloading, because all the firmware is sent first, and the Flash is erased and written with the new firmware only if all the specified checks are successful. In fact, there could be a power cut or reset during the brief Flash writing operation: in that case control of the Control unit is lost.

"Block" downloading

This is possible from Ver. 01.01.00 of the Control unit.
"Block" downloading offers the following two facilities:

1. downloading voluminous firmware to a Control unit without an adequate RAM for temporary storage of the entire file (therefore "block" downloading is essential for Cobra1500).
2. starting the communication as if it were a normal questioning session (e.g.: Polling if in /P) in order to warn that a download of firmware will follow.

Downloading takes place by starting from the last block. Whenever a 64Kbyte page is filled, the corresponding page on the Flash is updated and then downloading resumes until the process is completed.
With the "file2com" program, the length of each single block can be selected (e.g.: 8192 Byte).
There is no optimal value: the rule is that, in the absence of disturbance, the longer the block, the faster the download.
In the case of transmission channels subjected to disturbance (e.g.: via Modem), selection of short blocks (e.g.: 512 bytes) is preferable, because if a disturbance corrupts a block, re-transmission of that block would be briefer.
The following table illustrates a block download (blocks of 8 Kbyte each) of a file with a length of 166 Kbyte:

166KB (file length)																			
Page 1 (64KB)								**Page 2 (64KB)**								**Page 3 (38KB)**			
8K	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**8K**	**6K**
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**

- Note that block 21 is shorter than the others: (8K x 20) + 6K = 166K
- Block No. 21 is sent first, then block 20, then through to 17, which completes the uppermost page (Page 3)
- The data are temporarily stored in the RAM
- The contents of Page 3 in Flash are cleaned
- The RAM data (Page 3) are copied in the corresponding page in Flash.
- Downloading of firmware resumes from block 16 to 9, to write Page 2 (with the same mechanism as Page 3).
- Downloading of firmware resumes from block 8 to 1, to write Page 1 (with the same mechanism as Page 3 and 2).

N.B.: by its very nature, block downloading cannot guarantee 100% success of the firmware updating operation. In fact, if there is a power cut or a communication problem when at least one page has been written, control of the Control unit is lost.

One **very dangerous** situation could be as follows: unloading via Terminal Server while **LAN is heavily loaded** by other processes.
In case of doubt, you should opt for a **direct** local update of firmware via a portable PC.

Loss of control of the Control unit due to unsuccessful downloading.
To recognise this situation, power down and then power up the board; if downloading was unsuccessful:

- the board continues resetting (see relevant LED) about every three seconds.
- the display is not being managed (either blank or with meaningless characters).

Return the board to the **producer** to re-load the application program.

COMs used and type of downloading
Firmware download is possible by using either the Control unit's COM1 or COM2 port.
The following table summarises the possible alternatives and the situations in which they are used.

Port	Type of download	Control unit mode	Baud	Use and notes
COM1	Blocks	Operational /P	Set	Download from a PC to a remote pre-connected Control unit (RS485, RS232) is most convenient. Downloading possible from: - Wincontrol program - File2com
COM1	Blocks	Operational /T	Set	Download from a PC to a remote pre-connected Control unit (RS485, RS232, via Terminal Server, via Modem) is most convenient. Downloading possible from: - Wincontrol program - File2com
COM1	Linear			Linear download on COM1 not possible.
COM1		Keyboard programming		No downloading at all possible on COM1 if in Programming mode.
COM2 Aux	Blocks or Linear	Operational	57,600	If printing of "OK cards" is disabled for both A and B. A portable PC + cable is required for an on-site download. Downloading possible with the "file2com" program.
COM2 Aux	Blocks or Linear	Operational	Set	If "OK cards" printing is enabled for A or for B. (**DO NOT** do it during Keyboard programming).
COM2 Aux	Blocks or Linear	Keyboard programming	57,600	A portable PC + cable is required for an on-site download. Downloading possible with the "file2com" program.

Loading the "FILE2COM" program on the PC.
Only for those not using the Wincontrol program directly.
Create a "Folder" named "file2com". This name is not obligatory: if using another name, remember to use that name in the file search paths of the following examples.
Copy the following support files in that "folder":

- file2com.exe
- file2com.cfg
- file2com.hlp
- file2com topic.hlp
- file2com.cnt
- invio-16.inv
- Fmw-01-00-02.bin (this is the file to transfer: it could have a different name according to version).

N.B.: during use, the Log file, named **file2com.log**, is gradually generated and expanded.

Connection between PC and "Control unit"
New firmware updates are possible as from the 01.01.00 version – there are two operating options:

1. by exploiting the system connections (Polling, Modem, Terminal-server) for updates effected on the PC where the Wincontrol program operates.
2. by using a portable PC with an RS232 cable, and "visiting" every Control unit to be updated.

In case 1), which is obviously the most convenient, activate 'Block download', using the COM1 serial port on the Control unit.
Use case 2) in case of need or when point 1) is not feasible. Use the COM2 serial port on the Control unit.

If you use the system connections (case 1), there is no need to connect any other cables: the connection normally used in operating mode is sufficient. On the Wincontrol program PC, the software could be as follows:

- Wincontrol program
- file2com

If operating locally with a portable PC, use the cable as described in this paragraph. Connect it between a free serial port on the PC (e.g.: COM2) and the COM2-AUX port of the Control unit.
We mentioned a Portable PC, however, apart from the added inconvenience, a normal PC will do just as well.
Use the 'file2com' program.

Procedure for multiple updates via RS485, using a PC.

Multiple updates mean that the Control units must be updated one after another in sequence, repeating the operation several times and modifying each time on the PC the address of the panel No. being downloaded
Several Control units cannot be updated simultaneously by one download operation only.
The Control units, which must be **configured in /P**, are connected to each other and are linked to an RS232/RS485 interface.
Procedure with the **Wincontrol program** :

- operate from the Wincontrol program
- during the entire operation, the other Control units go into Stand-alone mode as Polling is interrupted
- activate the button with an Eprom icon and red arrow on the "**Panel Configuration**" menu
- select the number of the panel whose firmware has to be updated
- select the name of the file of the firmware to be downloaded (fmw-…-.bin)
- select "Execute"

Procedure with **File2com**:

- operate from the PC where the Wincontrol program is being executed
- exit the program; this means that, during the whole operation, all the Control units switch over to Stand-alone mode
- execute the "file2com" program
- during the entire operation, the other Control units go into Stand-alone mode as Polling is interrupted
- activate the "**Send in Blocks**" box
- recommended length of each block: 8,192 byte. If you think the line is particularly disturbed, it is best to lower this value.
- select the following Baud Rate related values: 57.600, N, 8, 1
- select "Panel options"
- specify the "Panel No." of the Control unit to be updated
- select the "Polling RS4485" box (the values in the other boxes are no longer significant)
- confirm with the "OK" push-button to return to the main mask
- indicate the name of the binary file containing the firmware update (by using the "…" button, you can browse and look for the file you require without typing in its name).
- select "Send file"

Procedure for multiple updates via Terminal-Server in LAN Ethernet (TCP/IP) using a PC.

Multiple updates mean that the Control units must be updated one after the other in sequence, changing the panel No. every time. Several Control units cannot be updated simultaneously by one download operation only.
The Control units, which **must be configured in /T**, are connected to a LAN Ethernet (TCP/IP) via a Terminal-Server.
Procedure with the **Wincontrol program** :

- operate from the Wincontrol program
- during a download via TCP/IP, the Wincontrol program continues to operate (polling, communication to other Control units configured in /T)
- activate the button with an Eprom icon and red arrow on the "**Panel configuration**" menu
- specify the "Panel No." of the Control unit to be updated
- select the name of the file of the firmware to be downloaded (fmw-…-.bin)
- select "Execute"

Procedure with **File2com**:

- operate from the PC where the Wincontrol program is being executed
- exit the program; this means that, during the whole operation, all the Control units switch over to Stand-alone mode

- execute the “file2com” program
- activate the “**Send in Blocks**” box
- recommended length of each block: 8,192 byte. If you think the line is particularly disturbed, it is best to lower this value.
- ignore the Baud Rate related values
- select “Panel options”
- specify the “Panel No.” of the Control unit to be updated
- select the “Via network (TCP/IP)” box (the values in the other boxes are no longer significant)
- indicate the TCP/IP address and the Terminal-Server port interfaced to the Control unit
- confirm with the “OK” push-button to return to the main mask
- indicate the name of the binary file containing the firmware update (by using the “…” button, you can browse and look for the file you require without typing in its name).
- select “Send file”

Procedure for multiple updates via Modem, using a PC.

Multiple updates mean that the Control units must be updated one after the other in sequence, changing the panel No. every time. Several Control units cannot be updated simultaneously by one download operation only.
The Control unit **must be configured in /T**, and can be reached by a telephone call, using a Modem.
Procedure with the **Wincontrol program** :

- operate from the Wincontrol program
- during a download via Modem, the Wincontrol program continues to operate (polling, communication to other Control units configured in /T)
- activate the button with an Eprom icon and red arrow on the “**Panel configuration**” menu
- specify the “Panel No.” of the Control unit to be updated
- select the name of the file of the firmware to be downloaded (fmw-…-.bin)
- select “Execute”

Procedure with **File2com**:

- operate from the PC where the Wincontrol program is being executed
- exit the program; this means that, during the whole operation, all the Control units switch over to Stand-alone mode
- execute the “file2com” program
- activate the “**Send in Blocks**” box
- recommended length of each block: 8,192 byte. If you think the line is particularly disturbed, it is best to lower this value.
- select the following Baud Rate related values: 19,200, N, 8, 1
- select “Panel options”
- specify the “Panel No.” of the Control unit to be updated
- select the “Via Telephone” box (the values in the other boxes are no longer significant)
- select the “Use Modem” box
- indicate the telephone No. of the Modem interfaced to the Control unit
- specify the Modem initialisation parameters. e.g.: ~~ATE0 V1X3 S0=0 S6=2^m~~
- confirm with the “OK” push-button to return to the main mask
- indicate the name of the binary file containing the firmware update (by using the “…” button, you can browse and look for the file you require without typing in its name).
- select “Send file”

Procedure for individual update via RS232, using the COM1 port for Polling

The Control unit **must be configured in /P.**
Procedure with the **Wincontrol program**:

- operate from the Wincontrol program
- get a cable identical to the one specified for Polling in RS232
- Connect this cable between a local portable PC and the Control unit’s COM1 port (the one with 4 pins)
- during the entire operation, the other Control units go into Stand-alone mode as Polling is interrupted and as, in any case, the units are physically disconnected.
- activate the button with an Eprom icon and red arrow on the “**Panel configuration**” menu
- specify the “Panel No.” of the Control unit to be updated
- select the name of the file of the firmware to be downloaded (fmw-…-.bin)

- select “Execute”

Procedure with **File2com**:

- Disconnect from the Control unit any connectors which may make it communicate with the Centre
- get a cable identical to the one specified for Polling in RS232
- connect this cable between a local portable PC and the Control unit’s COM1 port (the one with 4 pins)
- exit the Wincontrol program; this means that, during the whole operation, all the Control units switch over to Stand-alone mode
- execute the “file2com” program
- activate the “**Send in Blocks**” box
- recommended length of each block: 8,192 byte. If you think the line is particularly disturbed, it is best to lower this value.
- select the following Baud Rate related values: 57,600, N, 8, 1
- select “Panel options”
- specify the “Panel No.” of the Control unit to be updated
- select the “Polling RS232” box (the values in the other boxes are no longer significant)
- confirm with the “OK” push-button to return to the main mask
- indicate the name of the binary file containing the firmware update (by using the “…” button, you can browse and look for the file you require without typing in its name).
- select “Send file”

Procedure for individual update via RS232, using the Auxiliary COM2 port

The Control unit can be **configured either in /P or in /T**, because **COM2** is being used (and not COM1).
The Wincontrol program cannot be used

Procedure with **File2com**:

- get a cable as described in this paragraph
- connect this cable between a local portable PC and the Control unit’s Auxiliary COM2 port (the one with 5 pins)
- execute the “file2com” program
- activate the “**Send in Blocks**” box
- recommended length of each block: 8,192 byte. If you think the line is particularly disturbed, it is best to lower this value.
- select the following Baud Rate related values: 57,600, N, 8, 1
- select “Panel options”
- specify the “Panel No.” of the Control unit to be updated
- select the “Auxiliary RS232” box (the values in the other boxes are no longer significant)
- confirm with the “OK” push-button to return to the main mask
- indicate the name of the binary file containing the firmware update (by using the “…” button, you can browse and look for the file you require without typing in its name).
- if “Print OK cards” is activated, put the Control unit in **Programming Mode**
- select “Send file”

A **variation** of the previous case, but with **sending non in blocks** mode
Cobra1500 cannot be used, but **Cobra5000** can be, but only if the file to be downloaded is less than about 400 Kbyte.
If you have any doubts, you can, nevertheless, try to download: the download will either be successful, or the Control unit will abort the download before the given time because it has saturated the firmware reception buffer. In this case, send in blocks.
For Control unit with firmware version Ver. 01.01.00 or later, this is a simple alternative to the previous procedure (sending in blocks).
Important : sending in a non in blocks mode (i.e. “**Linear**”) is possible only with a **Cobra5000** and is the only way possible if using firmware **prior to Ver. 01.01.00.**

Procedure with **File2com**:

- get a cable as described in this paragraph
- connect this cable between a local portable PC and the Control unit’s Auxiliary COM2 port (the one with 5 pins)
- execute the “file2com” program
- **disable** the “Send in Blocks” box
- indicate the file named “**invio-16.inv**” (by using the “…” button, you can browse and look for the file you require without typing in its name).

- specify the characters for identifying an Rx Ok ("**OK**")
- select the following Baud Rate related values: 57,600, N, 8, 1
- ignore the "Panel options" settings
- indicate the name of the binary file containing the firmware update (by using the "…" button, you can browse and look for the file you require without typing in its name).
- if using Ver.01.01.00 or later, and if the "Print OK Cards" is activated, put the Control unit into **Programming Mode**; if the version is **previous** to Ver. 01.01.00, turn OFF and then turn ON the Control unit.
- select "Send file"

Whatever procedure you use, we shall now describe what happens after you select the "**Send file**" push-button.
If communication takes place (i.e. if the Control unit answers "OK"), the progress indicator bar gradually lengthens until the download is completed. The following message is shown on the 2 displays "NOW UPDATING FIRMWARE"
Wait for the Control unit to signal the outcome of the operation: this happens within about ten seconds.

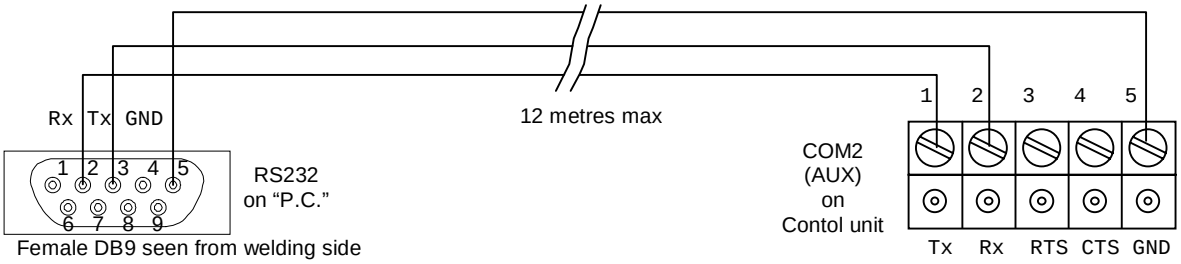
The procedure was **correctly** finished if:
1. the **4 LEDs** quickly lighted up several times one after the other in **sequence** (this is the Flash writing stage).
2. next, the 4 LEDs, the local Buzzer and the Buzzers on the readers are steadily and **simultaneously activated**.
3. after about 2 seconds, the Reset LED flashes briefly, all the signals of the previous point go OFF, and the Control unit begins to operate, using the new Firmware.

The procedure **was unsuccessful**, i.e. was terminated without any update, if:
1. this was missing: the stage when the **4 LEDs** are quickly lighted up several times one after the other in **sequence**
2. the 4 LEDs, the local Buzzer and the Buzzers on the readers are **simultaneously activated**.
   signalling is **pulsed** and attracts the operator's attention.
3. however, after about 2 seconds, the Reset LED flashes briefly, all the signals of the previous point go OFF, and the Control unit begins to operate, using the previous Firmware.

Making the connection cable for "Firmware update"

A cable for a 3-wire R232 is sufficient.
Only the following are used:
- Tx (Tx of the PC connected to the Control unit's Rx)
- Rx (Rx of the PC connected to the Control unit's Tx)
- GND (GND of the PC connected to the Control unit's GND)

Connect the Tx of a device to the Rx of the other. GND to GND.



We advise you to take a note and register the Control unit's operating parameters, because the current ones may be removed.
In fact, under certain conditions, the EEPROM (the one containing the operating parameters) is 'cleaned' and re-initialised with the standard parameters, which could be different from the current parameters due to subsequent modifications.
Re-initialisation occurs:
- if the first number of the FIRMWARE version has changed (from 01.00.00 to 02.00.00)

- if the second number of the Firmware version has changed (from 01.00.00 to 01.01.00)

There is **no** re-initialisation:
- if only the third number of the Firmware version has changed (from 01.00.02 to 01.00.03)