930 X1 / X1 Black
X2 / X2 Black

Manuale Utente

(r.2.6 - Ottobre 2018 – compatibile con le versioni di FW “g01_build2615” o “ap_a12_build454” e successive)

Nota: il contenuto può essere cambiato in ogni momento senza preavviso

AXESS TMC
a brand of Zucchetti Axess S.p.A

SOMMARIO

1. INTRODUZIONE 5
2. CARATTERISTICHE TECNICHE 6
3. INSTALLAZIONE 7
3.1 APERTURA E MONTAGGIO 7
3.1.1 X1 & X2 7
3.1.2 X1 & X2 Black 8
3.2 ALIMENTAZIONE, BATTERIE E COLLEGAMENTI PRINCIPALI 8
3.3 COLLEGAMENTO RELE' 9
3.4 COLLEGAMENTO INGRESSI DIGITALI 10
3.5 COLLEGAMENTO ETHERNET 10
3.6 COLLEGAMENTO LETTORI 11
3.7 LE SCHEDE DI ESPANSIONE OPZIONALI 914 NEOMAX 12
3.8 SELEZIONE LIVELLI DELLA PORTA SERIALE SU MORSETTIERA A VITE E COLLEGAMENTO STAMPANTE 13
3.9 VERSIONE HARDWARE 13
4. CONFIGURAZIONE 14
4.1 IMPOSTAZIONE DATA E ORA 16
4.2 ALARMS.TXT 18
4.3 HOLIDAYS.TXT 19
4.4 REASONS.TXT 19
4.5 FKEY.TXT E ENQUIRY.TXT 20
4.6 DIRECTION.TXT 22
4.7 READER1.TXT, READER2.TXT, EXTREADER.TXT 23
4.8 PRINTER.TXT, PRINTER_R*cc..cc*.TXT E PRINTER_Ennn.TXT 25
4.9 CONTROLLO REMOTO DEI RELE' E DEGLI INGRESSI DIGITALI DA WEB SERVER HTTP 26
4.10 IMPOSTAZIONE PARAMETRI 28
4.11 LISTA DEI PARAMETRI 30
4.12 ATTIVAZIONE DI FUNZIONI OPZIONALI DEL FIRMWARE 65
5. TABELLE DI CONTROLLO ACCESSI 68
5.1 FILE NECESSARI PER IL CONTROLLO ACCESSI 68
5.2 FILE OPZIONALI PER IL CONTROLLO ACCESSI 69
5.3 FORMATO DEI FILE PER IL CONTROLLO ACCESSI 70
5.4 CARDS.TXT 71
5.5 CARDRNGE.TXT 73
5.6 AUTHGRP.TXT 74
5.7 AUTH.TXT 75
5.8 TIMEMOD.TXT 75
5.9 USERS.TXT 77
5.10 AXREASON.TXT 79
5.11 CALENDAR.TXT 80
5.12 LA MODALITA' "SOLO PIN" 81
5.13 MESSAGGI DI ERRORE 82
5.14 CLOKI (IN PRECEDENZA DENOMINATO "WEB TABLE EDITOR") 84
6. GESTIONE AVANZATA DI UN VARCO 88
6.1 TIPO DI VARCO 88
6.2 TEMPI MASSIMI CONSENTITI PER IL PASSAGGIO 89
6.3 ASSEGNAZIONE DEGLI INGRESSI DIGITALI 90
6.4 ASSEGNAZIONE DELLE USCITE RELE' 92
6.5 GESTIONE ONLINE DEL VARCO 93
7. TRANSAZIONI 96
7.1 EMISSIONE E RIENTRO DI EVENTI RELATIVI ALLA GESTIONE AVANZATA DI UN VARCO 100
7.2 DEFINIZIONE DI UN FORMATO PERSONALIZZATO 101

7.3 INVIO DELLE TRANSAZIONI TRAMITE CLIENT FTP 102
8. LINGUE 105
9. AGGIORNAMENTO FIRMWARE 107
9.1 AGGIORNAMENTO DEL FIRMWARE DEI LETTORI 107
10. INTERFACCIA UTENTE DI X1/X2 108
10.1 AVVIO 108
10.2 STATO DI ATTESA (PRONTO AD ACCETTARE TRANSAZIONI) 108
10.3 DOPO UNA LETTURA DI CARTA, DIGITAZIONE DI CODICE O AUTENTICAZIONE BIOMETRICA 111
10.4 RICHIESTA CODICE PIN 112
10.5 MENU SUPERVISORE 113
10.6 TRANSAZIONI CON CODICE CAUSALE 116
10.7 REVISIONE DATI DI PRESENZA 118
10.8 MENU "RIDOTTO" PER SELEZIONE CAUSALI / *ENQUIRIES* REMOTE 119
11. IL MODULO BIOMETRICO ESTERNO FINGERBOX 120
11.1 MENU DI GESTIONE DELL'ARCHIVIO DELLE IMPRONTE 123
11.2 SALVATAGGIO DELLE IMPRONTE SU CARTE *MIFARE* 132
11.3 IMPORTAZIONE DI IMPRONTE NEL MODULO BIOMETRICO VIA FTP 134
11.4 ULTERIORI MODALITA' DI ESENZIONE DALLA VERIFICA BIOMETRICA 135
11.5 *ENROLLMENT* DISTRIBUITI SOTTO XATL@S 135
12. TRANSAZIONI ONLINE VIA HTTP 137
12.1 MESSAGGI HTTP PER TRANSAZIONI ONLINE (DA X1/X2 A MasterURL) 137
12.2 FORMATO RISPOSTA DEL SERVER (DA MasterURL A X1/X2) 139
12.3 MESSAGGIO "KEEP ALIVE" (DA X1/X2 A MasterURL) 140
12.4 FORMATO RISPOSTA DEL SERVER AL "KEEP ALIVE" (DA MasterURL A X1/X2) 141
12.5 MODALITA' ONLINE: SERVER NON IN LINEA 144
12.6 MESSAGGI HTTP ONLINE SU MODIFICA DELL'ARCHIVIO BIOMETRICO (DA X1/X2 A MasterURL) 145
13. FUNZIONAMENTO A BATTERIA 147
13.1 RICARICA RAPIDA DELLA BATTERIA 147
14. UTILIZZO DELLA CHIAVETTA USB 148
14.1 SCARICO TRANSAZIONI SU CHIAVETTA USB 148
14.2 SALVATAGGIO CONFIGURAZIONE SU CHIAVETTA USB 149
14.3 CARICAMENTO CONFIGURAZIONE DA CHIAVETTA USB 149
14.4 AGGIORNAMENTO FIRMWARE DA CHIAVETTA USB 150
14.5 SALVATAGGIO DATI BIOMETRICI SU CHIAVETTA USB 150
14.6 IMPORTAZIONE DATI BIOMETRICI DA CHIAVETTA USB 150
15. USO DEL MODEM GPRS OPZIONALE 152
15.1 VISUALIZZAZIONE STATO MODEM GPRS 155
16. ESECUZIONE DI COMANDI VIA FTP 156
17. TRATTAMENTO SICURO DEI DATI AI FINI DEL GDPR 157
17.1 DATI PERSONALI SUI TERMINALI 157
17.1.1 FILE DI LOG 157
17.1.2 FILE USERS.TXT 157
17.1.3 DATI BIOMETRICI 158
17.2 CRITTOGRAFIA SUI DATI PERSONALI 159
17.2.1 VERIFICA SUPPORTO CRITTOGRAFIA DA PARTE DEL TERMINALE 159
17.2.2 IMPOSTAZIONE DELLA CHIAVE DI CIFRATURA 159
17.2.3 VERIFICA CORRISPONDENZA CHIAVI DI CIFRATURA IMPOSTATE SU TERMINALE E SERVER 160
17.2.4 ATTIVAZIONE DELLA CRITTOGRAFIA 161
17.2.5 FORMATO DEI FILE CRIPTATI 161
17.3 HASH SU UTENZE E PASSWORD 162
17.4 HASH SU TRANSAZIONI ED EVENTI 163
17.4.1 ATTIVAZIONE DELL'HASH SU TRANSAZIONI ED EVENTI 163
18. LA VERSIONE SPECIALE PER SERRATURE WIRELESS OFFLINE DELLA SERIE APERIO 165

18.1 X1/X2 APERIO PLANT MANAGER BASIC PLANT CONFIGURATION 166
18.2 WIRELESS LOCKS INITIALIZATION 166
18.3 ACCESS CARDS MANAGEMENT 168
18.4 THE SPECIAL “APERIO” MENU 172
18.4.1 WIRELESS LOCKS INTERNAL CLOCK SETUP 172
18.4.2 SERVICE CARDS 173
18.4.3 CHECKING AND ERASING CARDS 174
18.4.4 OTHER OPTIONS 174
18.4.5 TROUBLESHOOTING 175
19. STRUMENTI SOFTWARE 176
20. MAPPE DEI CARATTERI 177

1. INTRODUZIONE

X1 e X2 sono gli innovativi terminali di Rilevazione Presenze e Controllo Accessi perfetti per tutte quelle situazioni in cui sono richiesti dispositivi compatti e robusti senza per questo compromettere funzionalità, tecnologia e design, grazie al loro ottimo rapporto prestazioni/prezzo.

Per chi già conosce i terminali AXESS TMC della generazione precedente, X1 e X2 sono differenti sia per come vengono configurati che per come comunicano.

Con X1 e X2 non sono necessari DLL e SDK proprietari, poiché lavorano con protocolli standard (HTTP e FTP) e file di testo standard.

Il protocollo TMC-UDP non viene usato con X1 e X2, con una sola eccezione per consentire una facile identificazione di tutti i terminali X1/X2 in rete (vedi §3.5 a pag. 10)

Su X1 e X2 il file system si trova su una micro-SD card rimovibile. La capacità della micro-SD card è di alcuni GB: ciò significa che è possibile registrare sulla memoria del terminale un numero enorme di transazioni e utenti autorizzati. Poiché tutte le transazioni, le tabelle di controllo accessi e i file di configurazione sono file di testo memorizzati nella micro-SD card, in caso di malfunzionamento del terminale, è sufficiente inserire la micro-SD del terminale guasto in un nuovo X1/X2, e l'applicazione host non si accorgerà neppure che il terminale è stato sostituito: l'unica cosa che cambia è l'indirizzo MAC.

Nel caso invece sia necessario sostituire la micro-SD card, è possibile copiare tutti i file dalla micro-SD originaria alla nuova per ripristinare l'esatta situazione precedente(*).

X1 e X2 sono configurabili mediante diversi parametri e tabelle, ma non sono programmabili in alcun modo (né tramite script, come con le PROC, né tramite programmazione in 'C' o .NET), pertanto le funzionalità già integrate non possono essere estese dall'utente.

Se dovete gestire transazioni complesse siete pregati di considerare i nostri terminali programmabili (TRAX, SuperTrax, Ultrax, …).

(*) Nota: la copia dei file dalla micro-SD originaria alla nuova deve essere effettuata in locale mediante un PC, secondo la procedura qui descritta:

1) create una cartella temporanea
2) inserite la micro-SD originaria nel PC e copiatene il contenuto nella cartella temporanea
3) inserite la nuova micro-SD nel PC e formattatela in FAT32
4) copiate il contenuto della cartella temporanea nella nuova micro-SD

La copia via FTP da un PC all'X1/X2 equipaggiato con la nuova micro-SD è vietata, e in ogni caso è sempre necessario formattare la nuova micro-SD in FAT32 prima di utilizzarla, anche partendo da una situazione pulita (cioè quando non si desidera o non è possibile recuperare i dati dalla micro-SD originaria).

2. CARATTERISTICHE TECNICHE

Tastiera	• 6 tasti funzione a membrana ai lati del display • Solo X2 & X2 Black: 10 tasti numerici a membrana (per inserire un codice causale o un PIN, o per operazioni di servizio)
Display	Transflettivo, garantisce un'eccellente visibilità anche in piena luce solare 128x64 a LED, bianco
Memoria	Almeno 4GB su micro-SD card interna. Nota: l'alloggiamento della micro-SD è accessibile solo aprendo il terminale
Lettore Primario	RFID integrato (125KHz o HID PROX in Clk&Data, Mifare o Legic o Hitag seriale, 125KHz + Mifare doppia testa in Clk&Data o seriale, HID PROX / iCLASS in Wiegand) o esterno magnetico Tk2 o Tk1/2/3 o barcode (Code39, Interleaved 2/5, EAN8, EAN13, Code128); è anche possibile gestire un lettore generico esterno con interfaccia Wiegand
Lettori Ausiliari	• Esterno (RFID Clk&Data 125KHz o HID / Mifare o Legic seriale, Magnetico Tk2 o Tk1/2/3, barcode) oppure, in alternativa, lettore di impronte digitali FingerBOX esterno su connettore molex (modalità 1:N e 1:1, 9590 impronte max) • Esterno (RFID Clk&Data 125KHz o HID / Mifare o Legic seriale, Magnetico Tk2 o Tk1/2/3, barcode) su connettore a vite estraibile con gestione di 2 LED; è anche possibile gestire un lettore generico esterno con interfaccia Wiegand o interfaccia seriale con livelli EIA-RS232. Utilizzabile solo in alternativa al modem GPRS opzionale • Fino a 2 lettori esterni aggiuntivi (RFID Clk&Data 125KHz, HID o Mifare, Magnetico Tk2) collegati ciascuno su una scheda di espansione opzionale 914 NeoMAX
Porte di Comunicazione	• Protocolli di comunicazione: TCP/IP, HTTP e FTP • Ethernet : 10/100 Mb/s PoE • 1 porta seriale su connettore a vite estraibile, impostabile su livelli TTL o RS232, per collegare un lettore esterno o una stampante, utilizzabile solo in alternativa al modem GPRS opzionale • Uscita Wiegand 37bit H10302 su connettore a vite estraibile per la ritrasmissione dei codici ricevuti dei lettori in locale (in alternativa alla gestione dei 2 LED di un lettore esterno) • 1 porta USB HOST 2.0 ad alta velocità per chiavette di memoria formattate in FAT32 • Disponibili versioni con modem GPRS opzionale (da richiedere all'acquisto del terminale), utilizzabile solo in alternativa al lettore ausiliario esterno su connettore a vite
Input/Output	• 1 relé interno (max 1A @ 30Vdc, carico resistivo): può essere usato per attivazioni temporizzate (sirene) o per sbloccare un varco di accesso • 2 ingressi digitali per contatti puliti (non è necessario fornire un'alimentazione), non utilizzabilii per conteggio di impulsi • Fino a 2 schede di espansione opzionali 914 NeoMAX, ciascuna con ulteriori 2 relé (max 1A @ 30Vdc) e 2 ingressi digitali (in totale, fino a 4 relé e 4 ingressi digitali aggiuntivi)
Alimentazione	PoE 802.3.af A&B compatibile o 10..48 Vdc
Batteria	4,8V 600mAh NiMh per 2 ore max di funzionamento continuo, con spegnimento automatico in caso di inattività
Interfaccia Software	Interfacciabile con il *middleware* Traxit32 e con il programma di controllo accessi Xatl@s
Caratteristiche Fisiche	Grado di protezione ambientale: IP55 Contenitore: ABS V0 (X1 & X2) oppure LURAN® SC (X1Black & X2 Black) Dimensioni: 120x130x52 mm (AxLxP) Massa: 350 gr Temperatura in funzionamento: -10…+50°C (la batteria non deve oltrepassare i 50°C)
Audio & Video	Segnalatore acustico multitono a volume regolabile su 3 livelli

3. INSTALLAZIONE

3.1 APERTURA E MONTAGGIO

3.1.1 X1 & X2

Per aprire X1 e X2 occorre prima rimuovere la cornice frontale, facendo leva sulle rientranze lungo i bordi superiore e inferiore (vedi vista dall'alto, figura 1).

Figura 1

Viti di chiusura agli angoli

Figura 2

E' quindi possibile svitare le 4 viti agli angoli 4 per sbloccare il frontale del terminale (vedi vista frontale, figura 2), in modo da poterlo tirare verso di voi perpendicolarmente alla sua superficie.

Supporti da forare per montaggio diretto a muro

Figura 3

A questo punto avete due scelte: potete fissare il retro del terminale direttamente al muro, forando almeno 2 dei 4 supporti plastici circolari (vedi vista posteriore, figura 3), oppure usare la staffa metallica opzionale che si inserisce nelle apposite scanalature sul retro (figura 4) e viene fissata mediante una singola vite.

Scanalature per staffa opzionale

Cappuccio in gomma porta USB

Figura 4

Il vantaggio principale di usare una staffa a muro è che si può facilmente rimuovere l'intero terminale dal muro e accedere a tutte le connessioni esterne attraverso la finestra posteriore, senza bisogno di aprire il contenitore.

In ogni caso, l'apertura del terminale è necessaria quanto meno per accedere al connettore della batteria principale (vedi §3.2 a pag. 8), e (solo se necessario) allo slot della micro-SD e alla batteria tampone dell'orologio, oltre che per controllare la versione hardware della scheda a circuito stampato (vedi §3.9 a pag. 13).

Una volta effettuate tutte le connessioni, si raccomanda di richiudere lo sportellino posteriore per mantenere la protezione IP55. Per fare passare i cavi potete rimuovere una o più sezioni della paretina plastica frangibile sul fondo della scatola.

3.1.2 X1 & X2 BLACK

Per fissare X1/X2 Black al muro è necessario usare la staffa metallica in dotazione, che si inserisce nelle apposite scanalature sul retro e si fissa con due viti laterali nella parte bassa (vedi figura 1).

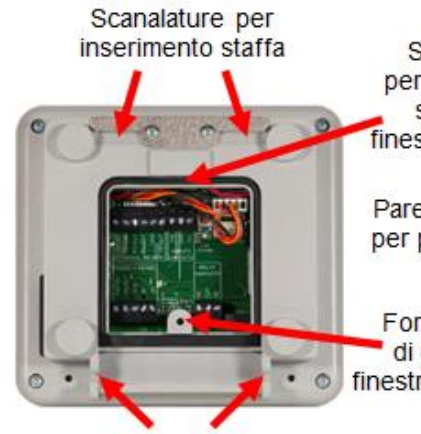


Figura 1

Figura 2

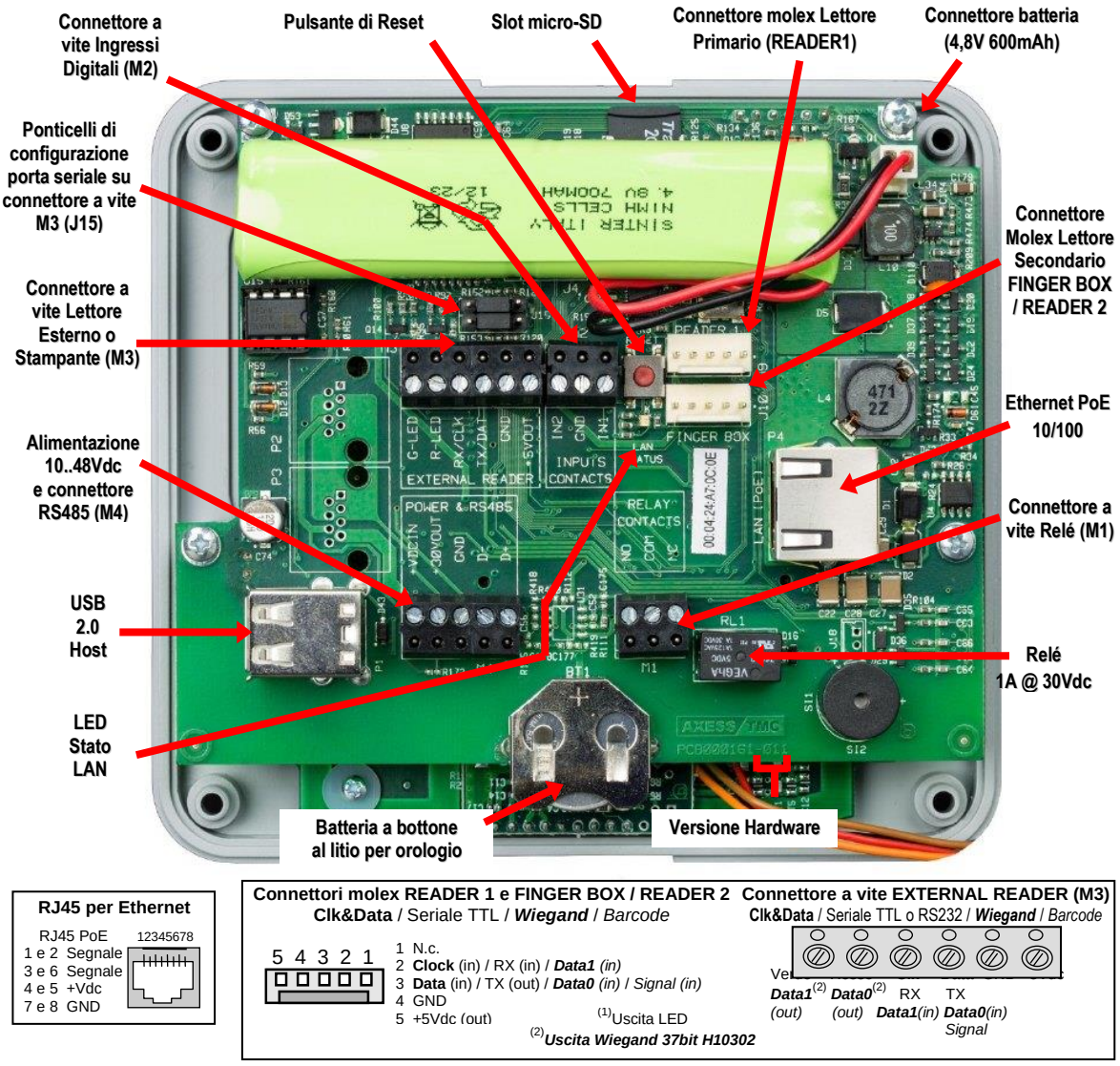
Figura 3

In questo modo si può facilmente rimuovere l'intero terminale dal muro e accedere a tutte le connessioni esterne rimuovendo solo lo sportellino posteriore (sempre figura 1), senza bisogno di aprire il contenitore: questo è necessario solo nel caso in cui si debba accedere allo slot della micro-SD. In tal caso, per aprire la scatola di X1 / X2 Black, dovete svitare le 4 viti ai 4 angoli sul retro del terminale (vedi figura 2). A questo punto potete fare leva sulle rientranze lungo i bordi superiore e inferiore del frontale del terminale (vedi figura 3), in modo da poterlo separare dal fondello tirandolo in direzione perpendicolare alla sua superficie.

Una volta effettuate tutte le connessioni, si raccomanda di richiudere lo sportellino posteriore per mantenere la protezione IP55. Per fare passare i cavi potete rimuovere una o più sezioni della paretina plastica frangibile sul fondo della scatola (vedi figura 2).

3.2 ALIMENTAZIONE, BATTERIE E COLLEGAMENTI PRINCIPALI

X1 e X2 possono essere alimentati sia con un adattatore 10..48 Vdc (che deve essere collegato ai morsetti **+VDCIN** e **GND** del connettore a vite estraibile **M4** – vedi figura della scheda – non funziona invertendo la polarità), sia tramite **PoE** (*Power over Ethernet*, IEEE802.3af), tipo A "*end-span*" (direttamente dallo switch) o tipo B "*mid-span*" (usando le due coppie del cavo Ethernet non utilizzate dai segnali dati). Controllate attentamente nello schema le etichette di tutti i collegamenti della morsettiera a vite e fate attenzione al corretto orientamento. E' qui mostrata la versione di hardware **011**.



Attenzione: X1 e X2 vengono forniti con le batterie scollegate e normalmente scariche, quindi la prima cosa da fare è collegare le batterie all'apposito connettore a 2 poli **J3**, situato nell'angolo in alto a destra sulla scheda.

Ad ogni modo, l'orologio integrato viene mantenuto da una batteria a bottone al litio. La batteria principale è ricaricata automaticamente quando l'alimentazione o il PoE sono collegati (vedi §13.1 a pag. 147). una ricarica rapida completa (disponibile solo a partire dalla versione di hardware **006**, vedi §3.9 a pag.13 e §13.1 a pag.147) può richiedere fino a 18 ore, mentre le batterie cariche possono avere un'autonomia massima di 2 ore in stand-by con un singolo lettore RFID 125KHz collegato ed in modalità display non retroilluminato.

Nota Importante: in caso X1/X2 debbano essere installati in ambienti ove la temperatura ambiente può superare i 40°C, si consiglia di posizionare le batterie di X1/X2 all'esterno del terminale. Oppure potete lasciare le batterie all'interno del terminale, scollegate: in questo caso dovete utilizzare un'unità UPS come sorgente di energia per gli alimentatori o gli switch PoE.

3.3 COLLEGAMENTO RELE'

X1/X2 dispone di 1 relé interno che può commutare un carico massimo di 1A @ 30Vdc, su entrambi i contatti normalmente aperto (**NO**) e normalmente chiuso (**NC**) sul connettore a vite estraibile M1 (vedi figura della scheda a pag. 9).

Inoltre, è possibile utilizzare fino a 4 relé aggiuntivi collegando fino a due schede di espansione 914 NeoMAX opzionali. Tali relé aggiuntivi saranno gestiti facendo riferimento ai numeri 2 e 3 per la scheda NeoMAX con indirizzo 1, e con i numeri 4 e 5

per quella con indirizzo 2, anche se su ciascuna scheda NeoMAX compaiono come R1 e R2 rispettivamente, e hanno le stesse caratteristiche del relé interno di X1/X2, sia per quanto riguarda il carico massimo che per la disponibilità di entrambi i contatti normalmente aperto e normalmente chiuso. Per la posizione dei contatti dei relé sulla scheda 914 NeoMAX fate riferimento al pieghevole allegato alla scheda stessa. Per dettagli sulla connessione fra X1/X2 e le schede 914 NeoMAX si veda il §3.7 a pag. 12.

Nota: raccomandiamo sempre di inserire, in parallelo ai contatti dei carichi induttivi (ad esempio serrature elettriche) e il più possibile vicino ad essi, un *varistore* (o *VDR*) da 50V per proteggere i circuiti da possibili sovratensioni.

3.4 COLLEGAMENTO INGRESSI DIGITALI

X1/X2 dispone di 2 ingressi digitali per contatti puliti utilizzabili esclusivamente per la logica di gestione avanzata del varco, vedi §6 a pag. 88, e che quindi non possono essere effettuare il conteggio di impulsi. Per attivare una linea di ingresso non è necessario fornire un'alimentazione, ma è sufficiente cortocircuitare il corrispondente pin IN1 o IN2 (sul connettore a vite estraibile M2, vedi figura della scheda a pag. 9) al pin comune GND, adiacente ad entrambi.

Inoltre, è possibile utilizzare fino a 4 ingressi digitali aggiuntivi collegando fino a due schede di espansione 914 NeoMAX opzionali. Tali ingressi aggiuntivi saranno gestiti facendo riferimento ai numeri 3 e 4 per la scheda NeoMAX con indirizzo 1 e con i numeri 5 e 6 per quella con indirizzo 2, anche se su ciascuna scheda NeoMAX compaiono come I1 e I2 rispettivamente, e come le linee di ingresso di X1/X2 possono essere usate per contatti puliti. Per la posizione dei contatti degli ingressi digitali sulla scheda 914 NeoMAX fate riferimento al pieghevole allegato alla scheda stessa. Per dettagli sulla connessione fra X1/X2 e le schede 914 NeoMAX si veda il §3.7 a pag. 12.

3.5 COLLEGAMENTO ETHERNET

Collegando il connettore RJ45, un effetto visibile sulla scheda è l'accensione del LED rosso **LAN STATUS** di controllo attività Ethernet (vedi figura della scheda a pag.9). Se lampeggia, significa che è stata rilevata attività di rete.

X1 e X2 vengono consegnati con DCHP abilitato, ma se il server DHCP non risponde, il terminale assume l'indirizzo IP di default 169.254.*x*.*y*, dove comunque *x* e *y* possono cambiare dinamicamente. In ogni caso, l'indirizzo MAC e le impostazioni IP correnti vengono mostrate all'accensione (vedi §10.1 a pag. 108), e possono essere facilmente modificate entrando nel menu supervisore (§10.5 a pag. 113).

E' anche possibile identificare facilmente da remoto tutti i terminali X1/X2 che abbiano già un indirizzo IP compatibile con la vostra sottorete, poiché essi rispondono ancora ad alcuni comandi Ethernet di basso livello (pacchetti di tipo "6" nel protocollo TMC-UDP) che ricevono sulla porta UDP 8499. Inviando questi comandi in modalità broadcast tutti i terminali AXESS TMC, inclusi X1 and X2, verranno trovati e identificati:

X → X1/X2 risponde inviando la configurazione IP corrente nel formato standard compatibile con EtherLite, con la prima linea che riporta parzialmente la versione fw nel formato "**V***NN***x**"

h → X1/X2 risponde inviando una stringa che rappresenta un MAC "esteso" di 12+4 cifre esadecimali, di cui le prime 12 rappresentano l'indirizzo MAC effettivo, mentre le ultime 10 rappresentano l'identificatore unico del terminale (per maggiori dettagli si veda il §4.12 a pag. 65)

V → si tratta di un nuovo comando TMC-UDP non supportato da tutti gli altri terminali AXESS TMC: X1/X2 risponde inviando la versione fw estesa e la data/ora del terminale nel formato "**X1 a***NN* **build** *nnn***,** *MMM gg aaaa hh*:mm:ss", dove *MMM* è una stringa costituita dai primi 3 caratteri del nome del mese in lingua inglese.

Inoltre, se il parametro **MasterURL** nella sezione *[Ethernet]* del file PARAMETERS.TXT (vedi §4.11 a pag. 56) viene impostato ad un indirizzo IP valido e raggiungibile X1/X2 invia anche spontaneamente a questo indirizzo ogni 60 secondi un messaggio TMC-UDP di tipo "*Keep Alive*", cioè un pacchetto di tipo "6" contenente il MAC "esteso" sopra descritto. Tale messaggio ha sempre come porta UDP sorgente la 8499, e tale valore ha anche la porta UDP destinazione a meno che non venga specificata una porta diversa aggiungendola all'IP all'interno del parametro **MasterURL**. In questo modo, X1/X2 può segnalare la sua esistenza e farsi identificare, ma in ogni caso non si aspetta nessuna risposta da parte del server.

Nota: la funzionalità "Keep Alive" via UDP è indipendente e si aggiunge senza sostituirla alla funzionalità "Keep Alive" via HTTP descritta al §12.3 a pag. 140: seppur apparentemente simili, quest'ultima ha uno scopo più complesso in quanto il terminale, oltre che per farsi identificare, la usa per sapere se il server è in linea (a seconda che riceva risposta oppure no), e per segnalare la presenza di eventuali timbrature registrate in modalità offline. Inoltre, mentre il protocollo TMC-UDP viene sempre gestito (limitatamente però ai soli 3 comandi sopra descritti), la gestione del protocollo HTTP è attiva solo se il parametro **Protocol** nella sezione *[Ethernet]* del file PARAMETERS.TXT (vedi §4.11 a pag. 56) viene impostato ad 1 (default 0).

3.6 COLLEGAMENTO LETTORI

Fino a 3 lettori di carte di "console" diversi possono essere collegati direttamente a X1/X2 (oppure, in alternativa, 2 lettori di carte e 1 modulo biometrico per il riconoscimento di impronte digitali FingerBOX).

L'eventuale lettore integrato all'interno del terminale è già collegato al connettore molex primario contrassegnato come "READER 1" (vedi figura della scheda a pag.9). Lo stesso connettore può essere utilizzato anche per collegare un qualunque tipo di lettore esterno, ad esempio un lettore magnetico o barcode a fessura che non può essere integrato nella scatola chiusa del terminale.

Un secondo lettore esterno può essere collegato al connettore molex secondario contrassegnato come "FINGER BOX" (o "READER 2" nelle versioni hardware fino alla 005, vedi §3.9 qui sotto), ma solo nel caso in cui tale connettore non venga già utilizzato per collegare l'apposito modulo di lettura impronte esterno FingerBOX, appunto. Si noti che eventuali LED presenti sul lettore esterno non potranno essere gestiti con questo tipo di collegamento.

Infine, un ulteriore lettore esterno può essere collegato alla morsettiera a vite estraibile **M3** contrassegnata come "EXTERNAL READER", ma solo nel caso in cui non sia già presente il modem GPRS opzionale (vedi§15 a pag. 152). E' anche possibile gestire gli eventuali 2 LED verde e rosso del lettore tramite gli appositi pin G-LED e R-LED: G sta per *Green* (verde) e R per *Red* (rosso; Nota: a partire dalla versione di fw a09_build84, i pin G-LED e R-LED sul connettore a vite estraibile "EXTERNAL READER" vengono gestiti anche in caso di letture effettuate su "READER 1" o "READER 2"): il comportamento del LED rosso in modalità di attesa (normalmente acceso / normalmente spento) può essere impostato con il parametro **ReaderLeds** nella sezione *[ExtReader]* del file PARAMETERS.TXT (vedi §4.11 a pag. 48). In alternativa, gli stessi pin G-LED e R-LED (assieme al pin GND sullo stesso connettore a vite) possono essere usati per ritrasmettere automaticamente ogni lettura effettuata (su uno qualunque dei 3 lettori diversi collegati) ad un controller remoto, nel formato fisso Wiegand 37bit H10302 (vedi parametro **WiegandOutput** nella sezione *[ExtReader]* del file PARAMETERS.TXT al §4.11, pag.47).

In tutti i casi si raccomanda di seguire attentamente l'ordine dei pin riportato nello schema a pag. 9, compatibilmente con lo specifico formato di uscita dei dati di ciascun tipo di lettore.

Inoltre, fino a 2 lettori di carte aggiuntivi possono essere collegati a X1/X2 attraverso delle schede di espansione opzionali 914 NeoMAX (una per ciascun lettore aggiuntivo). ***Nota***: le letture effettuate su tali lettori aggiuntivi sono in ogni caso gestite come se provenissero dal lettore "EXTERNAL READER", e non vi è alcuna distinzione fra le letture provenienti da una o dall'altra scheda 914 NeoMAX. Per la posizione e la pinatura del connettore molex del lettore sulla scheda 914 NeoMAX fate riferimento al pieghevole allegato alla scheda stessa. Per dettagli sulla connessione fra X1/X2 e le schede 914 NeoMAX si veda il §3.7 a pag. 12.

Attenzione: il corretto collegamento fisico non è comunque sufficiente per ottenere una corretta decodifica delle letture effettuate: a tale scopo è anche necessario impostare i parametri **CardDecode** in ciascuna sezione *[Reader1]*, *[Reader2]* e *[ExtReader]*(*) del file PARAMETERS.TXT (vedi §4.11 a pag.42) ad un valore compatibile con lo specifico formato di uscita dei

dati di ciascun tipo di lettore. Vedi anche il §3.8 a pag. 13. Per testare il funzionamento del lettore potete usare l'apposito pulsante "**Test Reader**" presente in ciascuna delle 3 omonime sezioni del web server HTTP relative ai lettori (vedi §4 a pag. 14; per comodità in ciascuna sezione è anche evidenziata la posizione del connettore corrispondente nello schema della scheda a circuito stampato): vi verrà chiesto a quel punto di effettuare una lettura sul lettore selezionato (pure qui, nel caso di "EXTERNAL READER", la lettura può anche essere effettuata su uno qualunque dei lettori aggiuntivi su schede 914 NeoMAX), dopodiché verrà mostrato il codice utente estratto in base all'attuale configurazione dei parametri **CardCodeBegin** e **CardCodeLength** ("*Code read*") e l'intero codice decodificato ("*RAW data*") in base all'attuale valore del parametro **CardDecode**. ***Nota***: dalla versione di firmware a07_build832 tutti i caratteri alfanumerici vengono accettati all'interno dei codici utente in seguito a lettura di carte per le transazioni di rilevazione presenze / controllo accessi.
Nel caso di collegamento di un modulo di lettura impronte esterno FingerBOX, invece, è necessario abilitarne la gestione impostando il parametro **Enabled**=1 nella sezione *[Biometric]* del file PARAMETERS.TXT (vedi §4.11 a pag. 48) oppure, analogamente, spuntando la checkbox "**Enabled**" nella pagina "**Biometrics**" del web server HTTP e confermando col pulsante "**Save**". Una volta fatto questo, la pagina "**Reader 2**" del web server HTTP risulterà priva di opzioni, mostrando solo la scritta rossa "**Reader used by finger box**". Per maggiori informazioni si veda il §11 a pag. 120.

(*) Come detto la sezione *[ExtReader]* si applica anche ad entrambi i lettori aggiuntivi su schede 914 NeoMAX, con l'eccezione dei parametri **CardDecode** e **BaudrateReader**, che in tal caso vengono ignorati poiché la decodifica viene effettuata autonomamente dai 914 NeoMAX ed è fissa all'equivalente del valore '0' di **CardDecode**, cioè lettore di carte magnetiche in traccia 2 o altro tipo di lettore con uscita compatibile.

3.7 LE SCHEDE DI ESPANSIONE OPZIONALI 914 NEOMAX

E' ora possibile collegare fino a 2 schede opzionali 914 NeoMAX, ciascuna in grado di aggiungere un lettore di carte ai lettori "di console", e di espandere le caratteristiche di I/O di X1/X2 aggiungendo 2 relé (max 1A @ 30Vdc), entrambi con contatti sia normalmente aperto che normalmente chiuso, e 2 ingressi digitali già alimentati per contatti puliti, come già visto ai §3.3 e §3.4: in totale, quindi, si possono avere fino a 2 lettori di carte, 4 relé e 4 ingressi digitali aggiuntivi.

X1/X2 viene collegato alle schede 914 NeoMAX attraverso una linea RS485 (morsetti **D+**, **D-** e **GND** del connettore a vite estraibile **M4**, vedi figura della scheda a pag. 9; per la posizione dei segnali della linea RS485 e dei contatti dei relé sulla scheda 914 NeoMAX fate riferimento al pieghevole allegato alla scheda stessa). ***Nota***: i collegamenti della linea RS485 devono seguire uno schema a "BUS" (lo schema a "STELLA" non è consentito).

Nel caso in cui X1/X2 non sia alimentato in **PoE** (né di tipo A "*end-span*" né di tipo B "*mid-span*"), è possibile alimentare le schede 914 NeoMAX in parallelo con lo stesso alimentatore di X1/X2, collegandole anche al morsetto **+VDCIN** del connettore a vite estraibile **M4**. Se invece X1/X2 è alimentato in **PoE**, le schede 914 NeoMAX possono comunque essere alimentate direttamente da X1/X2 attraverso il morsetto **+30VOUT** del connettore a vite estraibile **M4**. E' comunque sempre possibile usare per le schede 914 NeoMAX degli alimentatori separati, evitando in questo caso di collegarle ai morsetti **+VDCIN** e **+30VOUT** del connettore a vite estraibile **M4**.

Nota: gli indirizzi RS485 delle schede di espansione 914 NeoMAX opzionali devono essere impostati (mediante l'apposito *DIP switch* su ciascuna di esse) ai valori fissi '1' e/o '2' affinché esse vengano correttamente rilevate (se sono presenti entrambe, gli indirizzi RS485 devono essere diversi fra loro). Inoltre, il parametro **EnableNeoMaxI/O** nella sezione *[AccessControl]* del file PARAMETERS.TXT (vedi pag. 37) deve essere impostato a '1' (default).

3.8 SELEZIONE LIVELLI DELLA PORTA SERIALE SU MORSETTIERA A VITE E COLLEGAMENTO STAMPANTE

Solo nel caso in cui non sia già presente il modem GPRS opzionale (vedi§15 a pag. 152), la morsettiera a vite estraibile **M3**, contrassegnata come "EXTERNAL READER", può essere usata per un collegamento seriale a 3 fili (morsetti **RX**, **TX**, **GND**) con un dispositivo a scelta fra un lettore di carte oppure una stampante (quest'ultima solo a partire dalle versioni **011** dell'hardware e **a09_build11** del firmware). A seconda di quale sia lo standard elettrico del dispositivo da collegare, è possibile impostare i livelli della porta seriale come **TTL** o **EIA RS-232** usando i ponticelli di configurazione **J15** controllatene la posizione nella figura a pag. 9). Qui a lato sono mostrate le combinazioni da impostare nei due casi.

Ponticelli di configurazione porta seriale su M3

Se si utilizza la porta per collegare un lettore seriale, è necessario impostare i parametri fondamentali per la comunicazione nella sezione *[ExtReader]* del file PARAMETERS.TXT (vedi §4.11 a pag.42): **CardDecode** deve avere un valore compatibile con il formato dati specifico del lettore seriale (valori da 30 a 43), e **BaudRateReader** deve essere impostato alla velocità di trasmissione dei dati del lettore (si veda anche il §3.6 a pag.11). ***Nota***: ricordate che un generico lettore seriale di terze parti viene correttamente gestito solo se trasmette in dati in formato ASCII standard, con un singolo carattere *<CR>* come terminatore.

Se si utilizza la porta per collegare una stampante, invece, è necessario abilitarne la gestione impostando il parametro **Enabled**=1 nella sezione *[Printer]* del file PARAMETERS.TXT (vedi §4.11 a pag. 59) oppure, analogamente, spuntando la checkbox "**Enabled**" nella pagina "**Printer**" del web server HTTP e confermando col pulsante "**Save**". La stessa sezione include anche il parametro **BaudRate**, avente lo stesso scopo dell'analogo parametro usato per un lettore di carte. Una volta abilitata la gestione della stampante, la pagina "**External Reader**" del web server HTTP mostrerà la scritta rossa "**Reader used by Printer**".

La gestione della stampante da parte di X1/X2 consiste nella stampa di uno scontrino in seguito ad ogni transazione valida di rilevazione presenze o controllo accessi (con o senza selezione di una determinata causale) oppure in seguito a semplice selezione di una *enquiry* locale (vedi 4.8 a pag. 25): lo scontrino ha un formato definito dalla risposta del server (per le transazioni trasmesse in modalità online, vedi §12.2 a pag. 139) o dal contenuto del file PRINTER.TXT, PRINTER_R*cc..cc*.TXT e PRINTER_E*nnn*.TXT (per le transazioni validate localmente, vedi §4.8 a pag. 25).

3.9 VERSIONE HARDWARE

La versione hardware di X1/X2 è serigrafata sul lato inferiore della scheda a circuito stampato (vedi figura a pag. 9, per controllarla è necessario aprire il terminale). La versione hardware può essere importante poiché alcune funzionalità del terminale sono supportate soltanto a partire da una specifica versione di hardware (oltre che in quelle successive). Ad esempio, la ricarica rapida della batteria (vedi §13.1 a pag.147)e la gestione della porta USB host (vedi §14 a pag. 148) sono supportate solo a partire dalla versione **006**.

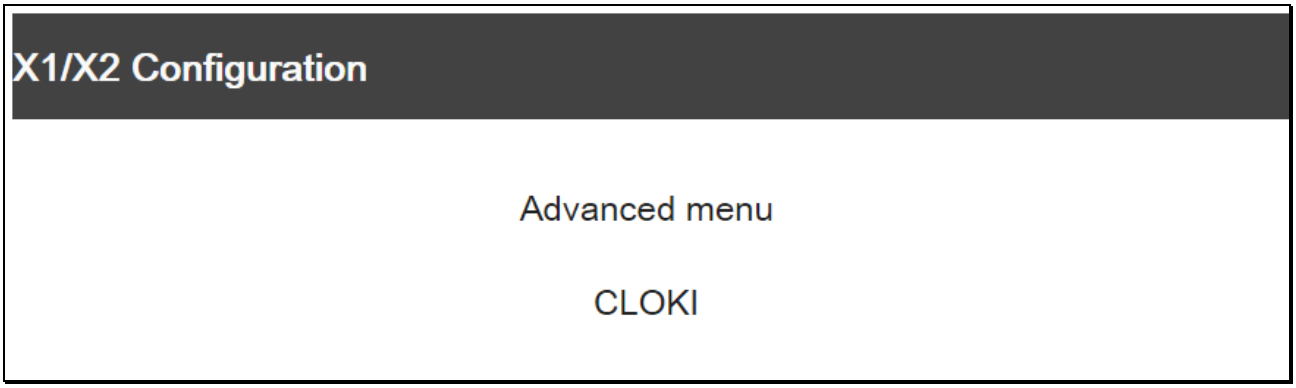
Avvertenza: raccomandiamo di non abilitare funzionalità non supportate dalla versione di hardware utilizzata (ove appositamente specificato), perché il risultato potrebbe essere il blocco di tutte o alcune funzioni del terminale. Ad esempio, evitate di abilitare la gestione della porta USB host (vedi §14 a pag. 148) se la versione di hardware non è almeno la **006**, altrimenti sarà necessario rimuovere la scheda SD e accedervi da un PC per reimpostare il valore di default e rendere di nuovo utilizzabile il terminale.

4. CONFIGURAZIONE

La configurazione può essere effettuata nei modi seguenti:

1) Caricando file di testo .TXT con qualunque programma client FTP (eccetto FireFox FireFTP) nella memoria del terminale
2) Collegandosi alla pagina iniziale del web server HTTP del terminale con qualunque browser standard(*)
3) Direttamente attraverso un programma server HTTP
4) Solo per alcune impostazioni: direttamente dal terminale (menu supervisore), vedi §10.5 a pag. 113

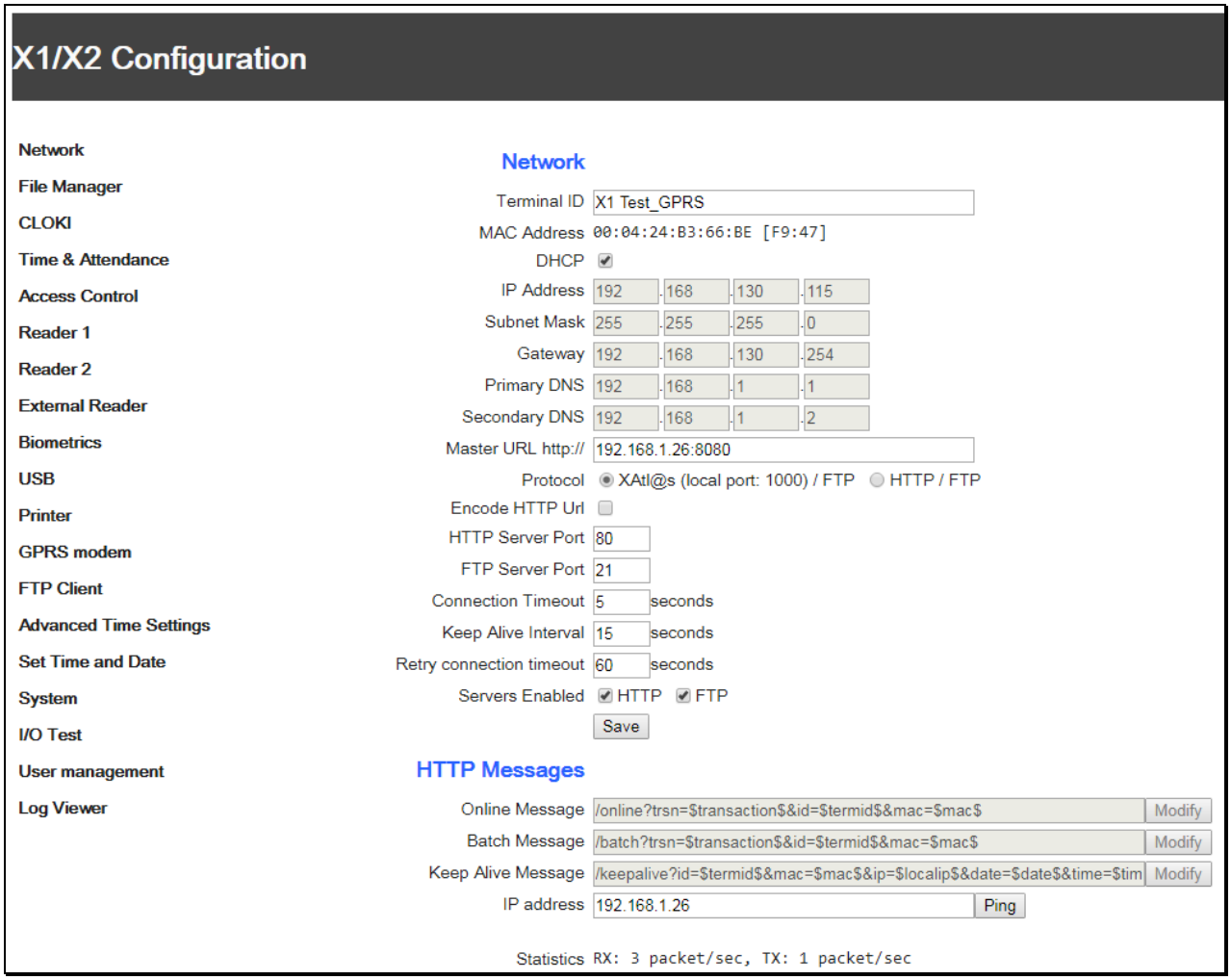
- Il metodo 1 è il modo principale per comunicare con il terminale da un programma: è sufficiente un client FTP per inviare file di testo di configurazione (.TXT), il cui formato predefinito è descritto nei parametri seguenti. L'unico account abilitato a tale operazione è quello amministratore (vedi sotto).
- Il metodo 2 è il più intuitivo dal punto di vista dell'utente, poiché consente di configurare il terminale mediante un menu a interfaccia grafica. Per accedere al web server è necessario inserire delle credenziali (nome utente e password), e il sistema prevede solo 2 possibili account utente: l'account amministratore, che consente la gestione completa del terminale (per il quale al default il nome utente è "admin" e la password è ancora "admin") e l'account per il solo accesso al web editor integrato CLOKI (vedi §5.14 a pag. 84), per il quale al default il nome utente è "manager" e la password è ancora "manager". Lo scopo di questa distinzione è consentire l'accesso alla configurazione avanzata (che include tutte le operazioni potenzialmente "pericolose", come cambiare i parametri di configurazione, cancellare file, formattare la memoria, ecc.) solo a determinati utenti (quelli con l'account amministratore), e lasciare invece solo la possibilità di modificare i file di servizio personalizzati (ad esempio le tabelle di controllo accessi, gli eventi temporizzati) e di visualizzare le timbrature effettuate ad altri utenti (quelli con l'account CLOKI). La pagina iniziale del web server (http://*<Indirizzo_IP_terminale>*) presenta pertanto due possibili opzioni, di cui la prima "**Advanced menu**" è selezionabile solo entrando con l'account amministratore (altrimenti si otterrebbe un messaggio di errore "Error 403 – Forbidden"), mentre il secondo "**CLOKI**" (precedentemente denominato " **Web table editor** ") può essere selezionato quando si accede sia con l'account amministratore che con l'account CLOKI.



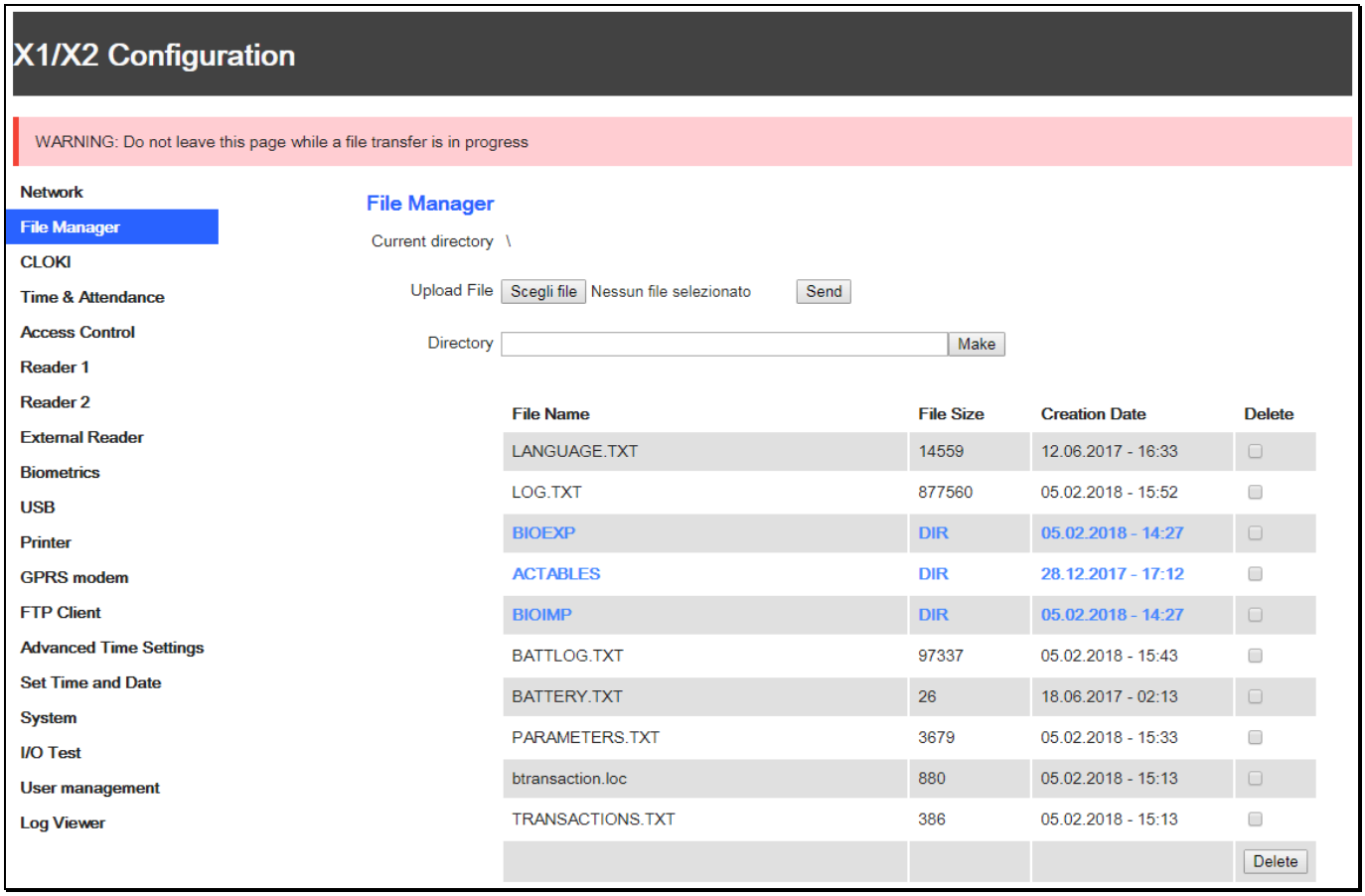
Attenzione: una volta effettuato un accesso con un account, l'autenticazione non viene più richiesta all'interno della medesima sessione di comunicazione del browser, quindi non è più possibile effettuare l'accesso con un account diverso (e quindi autorizzazioni diverse) senza prima chiudere il browser.

Una volta effettuato l'accesso al server HTTP con l'account "admin" e fatto click su "**Advanced menu**" verrete ridiretti sulla pagina "**Network**" del menu avanzato, vedi figura qui sotto. Usando il collegamento "**User Management**" sulla sinistra potrete poi modificare in modo indipendente sia il nome utente che la password per l'account amministratore (la cui modifica viene applicata sia per gli accessi in FTP che per quelli in HTTP), e per l'account CLOKI.

(*)***Attenzione***: il web server di X1/X2 dispone di un'interfaccia utente adattabile, vale a dire che gli elementi della pagina vengono distribuiti automaticamente in base alla dimensione della finestra al fine di migliorare la loro visibilità. Questa funzionalità funziona bene su tutti i browser moderni, ma se si utilizza Internet Explorer, assicurarsi che sia almeno la versione 9 e che la modalità "Vista compatibile" sia stata disattivata.



Nel seguito si assume sempre di avere già effettuato l'accesso al menu HTTP avanzato. Questo, inoltre, consente di inglobare anche il metodo 1 senza la necessità di usare un client FTP in una finestra separata, grazie alla sezione "**File Manager**": si può selezionare un qualunque file di testo salvato sul PC usando il pulsante "Sfoglia…" e caricarlo sul terminale via HTTP col pulsante "Send", oppure creare una nuova cartella sul terminale digitandone il nome e usando il pulsante "Make", oppure visualizzare il contenuto di un file già presente sul terminale (semplicemente facendo click sul relativo link nella lista; usate poi il pulsante "*indietro*" del vostro browser per tornare al menu principale) o scaricarlo su PC (facendo click col tasto destro sul link e poi selezionando "Salva oggetto/link con nome…"), e infine cancellare uno o più file spuntando le relative *checkbox* e infine premendo "Delete" (nota: le cartelle "**BIOEXP**" e "**BIOIMP**" evidenziate in arancione nella figura seguente vengono create solamente dopo avere abilitato la gestione dell'eventuale modulo biometrico esterno FingerBOX, come descritto al §3.6 a pag. 11, altrimenti non sono presenti):



Avvertenza: il contenuto di un file già visualizzato all'interno della medesima sessione di comunicazione del browser non viene più aggiornato automaticamente (neppure tornando al menu principale e selezionando nuovamente il file), anche se nel frattempo è cambiato. Usate **Ctrl**+**F5** sulla vostra tastiera per visualizzare sempre i dati aggiornati ed assicurarvi che la cache del browser sia stata svuotata.

- Il metodo 3 è un modo alternativo al 1 per automatizzare la configurazione da un programma, anche se la soluzione client HTTP dovrebbe piuttosto essere usata per ricevere transazioni in online (vedi §12 a pag. 137) e inviare risposte.

  La gestione dei messaggi con protocollo HTTP può essere attivata impostando a 1 il parametro **Protocol** nella sezione *[Ethernet]* del file PARAMETERS.TXT (vedi §4.11 a pag. 56) oppure, analogamente, selezionando il *radio button* "**HTTP**" fra le opzioni "**Protocol**" nella pagina "**Network**" del web server HTTP (vedi figura a pag. 15): questa modalità standard di comunicazione sostituisce il protocollo proprietario TMC-UDP utilizzato sugli altri nostri terminali.

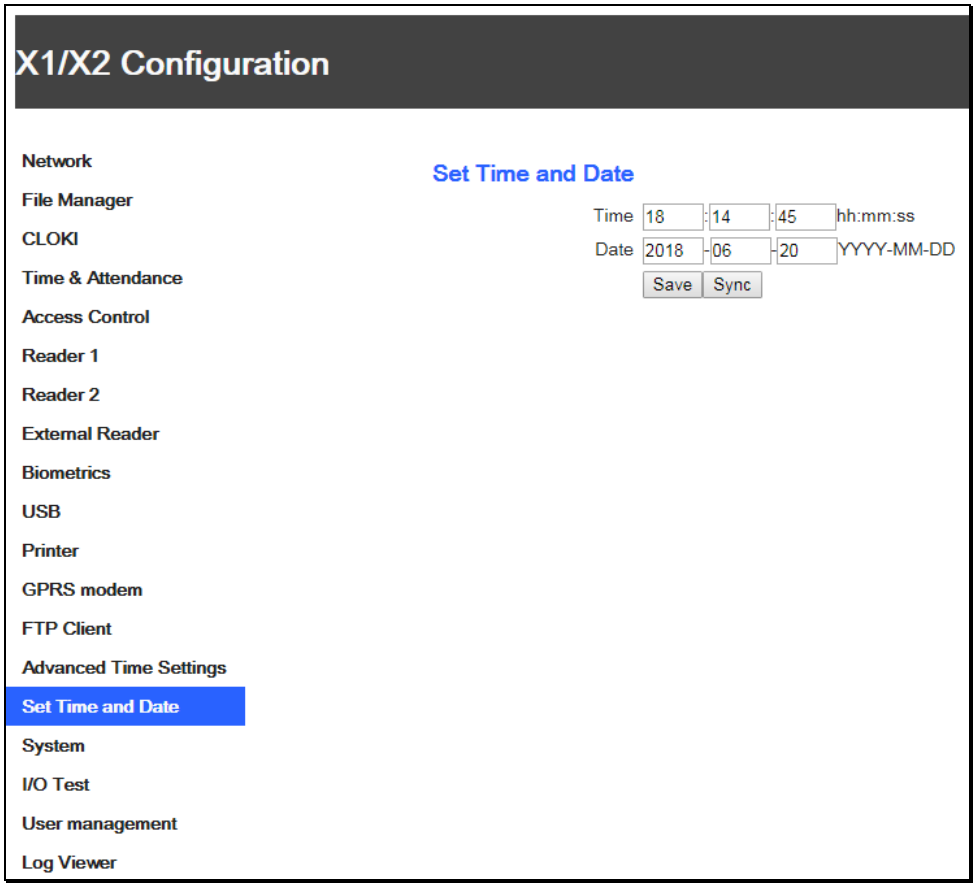
  Il valore di default del parametro **Protocol** (0) corrisponde invece al *radio button* "**XAtl@s / FTP**": viene infatti usato nel caso in cui X1/X2 venga gestito dal programma Xatl@s, oppure nel caso si intenda usare solo il protocollo FTP per configurarlo o scaricare in modalità *batch* le transazioni registrate in offline dal terminale (il protocollo FTP è comunque sempre attivo, anche se è stato selezionato il *radio button* "**HTTP**").

  Si veda il §12.3 a pag. 140 per approfondire il concetto di messaggio "*Keep Alive*" via HTTP, ed il §12.4 a pag. 141 per sapere quali devono essere il formato della risposta del client HTTP ed i comandi di configurazione disponibili.

4.1 IMPOSTAZIONE DATA E ORA

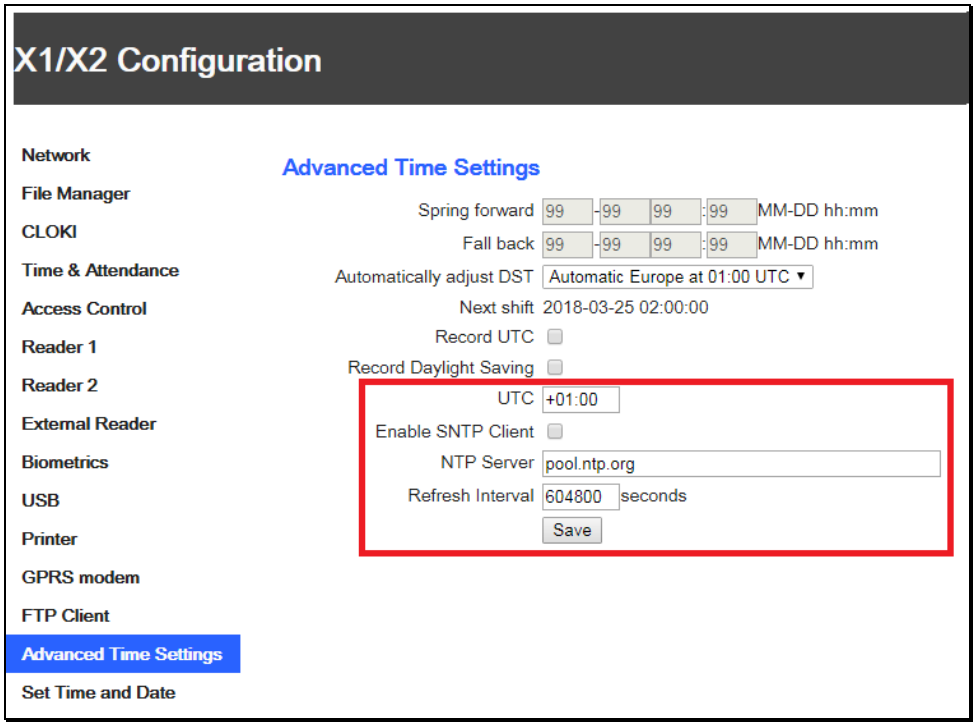
Metodo HTTP

Usando un browser web standard come FireFox o Internet Explorer, collegatevi alla pagina iniziale del web server di X1/X2, fate click sul link "**Set Time and Date**" sul lato sinistro e a quel punto o premete il pulsante "**Sync**" (che sincronizza l'orologio del terminale con quello del PC), o riempite i campi testo "**Time**" e "**Date**" con i valori che volete assuma il terminale:



Metodo SNTP

Impostando il parametro **UseNTP** all'interno della sezione *[TimeSettings]* del file PARAMETERS.TXT (vedi §4.11 a pag. 55) al valore 1 (default 0), oppure, analogamente, spuntando la checkbox "**Enable SNTP Client**" nella pagina "**Advanced Time Settings**" del web server HTTP, il terminale è in grado di aggiornarsi automaticamente, inviando una richiesta SNTP al server di sincronizzazione NTP specificato dal parametro **NTPServerName**, sempre all'interno della sezione *[TimeSettings]* del file PARAMETERS.TXT o della pagina "**Advanced Time Settings**" del web server HTTP. Tale stringa può essere un indirizzo IP o un URL logico (default "pool.ntp.org"): in quest'ultimo caso, ricordate che è necessario avere impostato gli indirizzi IP dei server DNS primario e secondario (parametri **Primary_DNS** e **Secondary_DNS** all'interno della sezione *[Ethernet]* del file PARAMETERS.TXT) da contattare per risolvere il nome dell'URL logico.



Se l'aggiornamento automatico della data/ora è abilitato, circa 30 secondi dopo ogni riavvio del terminale ed ogni aggiornamento dei parametri di configurazione viene mandata una richiesta di sincronizzazione SNTP. Se il server NTP specificato non è raggiungibile, il terminale rinnova la richiesta automaticamente ogni 30 secondi, fino a quando non ottiene risposta. A questo punto, se non vi sono ulteriori riavvii del terminale o aggiornamenti dei parametri di configurazione, il terminale effettuerà la successiva richiesta solo dopo l'intervallo impostato dal parametro **NTPRefresh**, sempre all'interno della sezione *[TimeSettings]* del file PARAMETERS.TXT o della pagina "**Advanced Time Settings**" del web server HTTP. Tale parametro è espresso in secondi (il default è 604800, che equivale a una settimana). Nota: l'intervallo minimo fra una richiesta e l'altra è di 30 secondi, pertanto qualunque valore del parametro inferiore a 30 ha in realtà lo stesso effetto del valore 30.

Attenzione: affinché l'orario venga impostato correttamente, dovete ricordarvi di impostare il parametro **UTC**, sempre all'interno della sezione *[TimeSettings]* del file PARAMETERS.TXT (vedi §4.11 a pag. 55) o della pagina "**Advanced Time Settings**" del web server HTTP, al valore relativo al vostro fuso orario (ad esempio per l'Italia dovete impostare il valore "+01:00").

Metodo con file di testo → **DATETIMEaaaammggHHMMSS.txt**

Se via FTP viene caricato un file avente un nome con questo formato, il terminale sincronizza immediatamente la sua data e ora con i valori contenuti nel nome del file.

Nota1: il contenuto del file è irrilevante, potrebbe anche essere vuoto.

Nota2: a partire dalla versione a08_build053, il file viene cancellato automaticamente dopo l'impostazione di data e ora. Se si è in possesso di un firmware antecedente, invece, il file non viene cancellato automaticamente, pertanto occorre effettuare la cancellazione del file subito dopo l'invio (sempre via FTP) per evitare l'accumularsi di numerosi file orari: consigliamo comunque di effettuare l'aggiornamento scaricando l'ultima versione di firmware disponibile seguendo le istruzioni del §9 a pag. 107.

E' a disposizione, su richiesta, un semplice file *batch* eseguibile da prompt dei comandi per effettuare l'impostazione di data e ora via FTP in maniera automatica su un terminale X1/X2 di cui viene specificato l'indirizzo IP.

Metodo manuale da menu supervisore

Nel caso in cui X1/X2 venga utilizzato come terminale *standalone* senza un collegamento Ethernet, i metodi descritti in precedenza non possono essere usati. E' comunque sempre possibile impostare l'ora manualmente sfruttando il menu supervisore del terminale, come descritto al §10.5 a pag. 113.

4.2 ALARMS.TXT

Lista delle attivazioni temporizzate dei relé e degli invii schedulati del file TRANSACTIONS.TXT corrente tramite client FTP (per maggiori dettagli vedi §7.3 a pag. 102.

Il formato di ogni record è il seguente:

HHMM,R,TTT,DLMMGVSF

Dove:

HHMM

HH=ore, MM=minuti.
Nota: è possibile utilizzare il carattere speciale '_' per rappresentare un qualsiasi valore del campo. Ad esempio, per impostare una schedulazione ad ogni inizio di ora, i campi HHMM avranno valore "__00"

R

Relé da attivare (1 è il relé interno, 2 e 3 sono riservati ai relé opzionali esterni, vedi §3.3 a pag. 9) *oppure*
"F" → attiva una connessione FTP client temporanea per inviare le transazioni ad un server
"G" → attiva una connessione / disconnessione GPRS schedulata
"!" → effettua un riavvio del terminale

TTT

Tempo di attivazione del relé:
"0" → il relé viene disattivato (da usare per terminare una precedente attivazione per un tempo indefinito)
"1".."254" → il relé viene attivato per il numero di secondi specificato
"255" → il relé viene attivato indefinitamente
oppure
"1" → se si vuole attivare una connessione FTP client temporanea per inviare le transazioni ad un server, o una connessione GPRS schedulata, questo campo deve essere impostato al valore fisso "1".
"0" → se si vuole schedulare una disconnessione GPRS automatica (opzionale: l'host HTTP può forzare la chiusura della connessione in qualunque momento usando il tag "**gprs=off**" tag nella risposta ai pacchetti "Keep Alive" che gli arrivano, vedi §12.4 a pag. 141), questo campo deve essere impostato al valore fisso "0".
oppure
irrilevante in caso di riavvio del terminale

DLMMGVSF

D=Domenica… S=Sabato e F=Festivi
Se >=1 → il relé viene attivato nel giorno corrispondente
Se <=0 → il relé non viene attivato nel giorno corrispondente

Nota: questo file può anche essere facilmente creato e modificato mediante la sezione "**TABELLE/ALLARMI**" dell'estensione firmware "CLOKI", se presente (vedi §5.14 a pagina 84).

4.3 HOLIDAYS.TXT

Lista delle date dei giorni festivi da considerare quando si usa ALARMS.TXT.

Il formato di ogni record è il seguente:

GGMM

Dove: GG=giorno del mese, MM=mese

Nota: questo file può anche essere facilmente creato e modificato mediante la sezione "**TABELLE/ALLARMI**" dell'estensione firmware "CLOKI", se presente (vedi §5.14 a pagina 84).

4.4 REASONS.TXT

Lista dei codici causali e loro descrizioni. X1/X2 è in grado di gestire un massimo di 40 causali, pertanto raccomandiamo di non inserire più di 40 record in questo file, poiché i record in eccesso verrebbero comunque ignorati.

Il formato di ogni record è il seguente:

cc..cc,descrizione

Dove *cc..cc* è il codice causale (che deve essere formato da sole cifre puramente numeriche; il numero di cifre può variare a seconda delle necessità: min 1, max 8)

Nota: è anche possibile definire delle causali personalizzate, cioè selezionabili solo da alcune tipologie di utenti. A tale scopo, è necessario abilitare il controllo accessi e caricare i file **CARDS.TXT**, **USERS.TXT** e **AXREASON.TXT** (vedi §5 a pag. 68). Si tenga presente che il file AXREASON.TXT (vedi §5.10 a pag. 79) è sempre più prioritario del file REASONS.TXT, indipendentemente dall'abilitazione del controllo accessi. Pertanto, se AXREASON.TXT è presente, REASONS.TXT non viene mai considerato (è come se non ci fosse).

Nota: questo file non può essere creato o modificato mediante la sezione "**TABELLE/CAUSALI**" dell'estensione firmware "CLOKI", se presente (vedi §5.14 a pagina 84), in quanto essa è specifica per l'utilizzo del file AXREASON.TXT.

4.5 FKEY.TXT E ENQUIRY.TXT

FKEY.TXT rappresenta la lista dei tasti numerici usabili per la selezione diretta (scelta rapida) della causale senza dover passare dal menu di selezione, vedi §10.6 a pag. 116 (solo sui modelli X2 con tastiera numerica). Si possono definire fino a 10 tasti di scelta rapida diversi (tanti quanti sono i tasti numerici sulla tastiera dell'X2).

E' anche possibile associare alcuni tasti numerici alla scelta rapida di particolari *enquiries* invece che alla scelta rapida di causali, nel caso in cui venga caricato anche l'apposito file ENQUIRY.TXT che ne contiene la lista. ***Nota***(**): se viene caricato il file ENQUIRY.TXT, premendo il tasto "freccia su" (▲) nella schermata di stand-by non verrà più automaticamente lanciata la procedura di revisione dei dati di presenza (vedi §10.7 a pag. 118), bensì verrà mostrato il menu di selezione delle *enquiries*, allo stesso modo in cui il tasto ""freccia giù" (▼) attiva il menu di selezione delle causali. L'unico modo per accedere alla revisione dati, in questo caso, è aggiungere al file ENQUIRY.TXT una enquiry con l'identificativo predefinito "99" (tipicamente in prima posizione), che corrisponde sempre alla revisione dati locale: la relativa descrizione, che verrà visualizzata nel menu delle enquiry, può essere scelta a piacere purché identifichi in maniera chiara la procedura di revisione dati di presenza. In questa situazione, quindi, per lanciare la revisione dati è necessario premere il tasto "freccia su" (▲), selezionare la voce di menu di enquiry relativa alla revisione dati locale (solo nel caso in cui tale voce sia la prima della lista, allora sarà già automaticamente selezionata) e confermare con il tasto ↵ (Enter).
Se X1/X2 viene controllato dal programma Xatl@s, le *enquiries* possono essere gestite da remoto (in questo caso, comunque, il file FKEY.TXT viene caricato automaticamente e non deve essere modificato).
In alternativa, ma solo se X1/X2 è collegato ad una stampante (vedi §3.8 a pag. 13), alcune *enquiries* possono anche essere gestite in locale, esclusivamente per effettuare delle stampe di scontrini (contenenti solo totalizzatori come dati variabili, ma non dati personali): in questo caso l'identificativo delle *enquiries* locali di stampa contenute nel file descrittivo ENQUIRY.TXT, vedi sotto, deve essere un numero *nnn* >= 100 per distinguerle da quelle gestite da remoto; gli stessi identificativi di *enquiries* vengono usati come riferimento nei nomi dei file di definizione del formato degli scontrini da stampare "PRINTER_E*nnn*.TXT", vedi §4.8 a pag. 25). ***Nota***: le *enquiries* locali di stampa, diversamente da quelle gestite da remoto, non possono essere lanciate premendo direttamente il corrispondente tasto numerico sulla tastiera fisica dei modelli X2, ma anche se definite nel file FKEY.TXT possono comunque essere selezionate solo dal menu di selezione "ridotto" accessibile premendo il tasto ↵ (Enter) nella schermata di stand-by, come descritto più in basso, oltre che dal menu completo di selezione delle *enquiries* accessibile premendo il tasto "freccia su" (▲) nella schermata di stand-by.

E' anche possibile associare due tasti numerici qualunque alla selezione diretta della direzione di passaggio (un tasto per l'ENTRATA e uno per l'USCITA, selezionabili a prescindere dall'eventuale direzione preimpostata dal terminale), da effettuare subito prima della lettura di un badge. E' anche possibile configurare il terminale in modo che non sia più possibile effettuare una transazione senza prima avere effettuato una selezione diretta della direzione o di una causale a scelta rapida, impostando il parametro **MandatoryFunction**=1 nella sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag.

34) oppure, analogamente, spuntando la checkbox "**Mandatory Function Key**" nella pagina "**Time & Attendance**" del web server HTTP e confermando col pulsante "**Save**".

Infine, è anche possibile associare un tasto numerico qualunque alla selezione della modalità digitazione manuale di causali numeriche "libere" (ovvero non predefinite in REASONS.TXT né in AXREASON.TXT) prima di effettuare la lettura di tessera (vedi anche par. **AllowTypeReason** nella sezione *[TimeAttendance]* del file PARAMETERS.TXT al §4.11 a pag. 35).

In ogni caso, le descrizioni delle sole causali e/o delle *enquiries* e/o delle direzioni di passaggio associate a tasti numerici per la scelta rapida sono anche mostrate in un menu di selezione "ridotto" (per come è definito può avere al massimo 10 righe) che può essere visualizzato dalla schermata di stand-by (attesa lettura carta) premendo il tasto ↵ (Enter), vedi §10.8 a pag. 119. Ecco perché il file FKEY.TXT può comunque essere utilizzato anche sui modelli X1 che non dispongono della tastiera numerica.

- Il formato del file FKEY.TXT è il seguente:

  [Functions]
  F*N*=**R***cc..cc*
  …
  F*N*=**E***n*
  …
  F*N*=**D***v*
  …
  F*N*=**N**
  …

La prima riga **[Functions]** è fissa, e ad essa seguono uno o più record (max 10) aventi ciascuno uno qualunque dei 4 formati mostrati, dove i caratteri in grassetto '**F**', '**=**', '**R**', '**E**', '**D**', '**N**' sono fissi (quest'ultimo viene usato per associare ad un tasto la funzionalità fissa di digitazione manuale di causali numeriche "libere"), mentre:

N è il numero del tasto numerico (1..9, 0)

cc..cc è il codice di una causale, che deve essere contenuto nel file descrittivo attualmente utilizzato, che può essere REASONS.TXT (vedi §4.4 qui sopra) oppure AXREASON.TXT, se presente (vedi §5.10 a pag. 79). Nel caso in cui si tratti di AXREASON.TXT, si noti che se una causale è disabilitata, il corrispondente tasto di scelta rapida viene ignorato, esattamente come accade per la visualizzazione nel menu di selezione standard della causale e nel menu di selezione "ridotto" per causali / *enquiries*.

n è il numero ordinale di una *enquiry* (quindi non il suo identificativo) all'interno del file descrittivo ENQUIRY.TXT (vedi sotto). Se X1/X2 viene controllato dal programma Xatl@s, tale file viene caricato automaticamente e non deve essere modificato.

v è una direzione di passaggio (0 sta per USCITA, 1 sta per ENTRATA).

Nota: nel caso in cui siano abilitate le modalità "digitazione manuale del codice tessera" oppure "solo PIN" (vedi §10.2 a pag. 108), non è più consentita la scelta rapida tramite tasto numerico, pertanto il file FKEY.TXT viene usato esclusivamente per definire il menu di selezione "ridotto".

Nota: questo file può anche essere facilmente creato e modificato mediante la sezione "**TABELLE/FKEY**" dell'estensione firmware "CLOKI", se presente (vedi §5.14 a pagina 84).

- Il formato di ogni record del file ENQUIRY.TXT è il seguente:

  nnn,*descrizione*

Dove *nnn* è l'identificativo della *enquiry*: i valori da 1 a 98 identificano le *enquiries* remote, il valore 99 identifica la procedura standard di revisione locale dei dati di presenza (vedi §10.7 a pag. 118), i valori >=100 identificano le *enquiries* locali di stampa.

Esempio di FKEY.TXT che associa i due tasti numerici 1 e 5 alla selezione forzata delle direzioni ENTRATA e USCITA, i tasti 2 e 3 ai codici causali 11 e 22 e il tasto 4 alla prima enquiry descritta all'interno del file ENQUIRY.TXT, ovvero l'enquiry remota con identificativo 10; inoltre consente la visualizzazione all'interno del menu di selezione "ridotto" della seconda e terza enquiries descritte all'interno del file ENQUIRY.TXT, ovvero le enquiries locali di stampa con identificativi 110 e 120, che si riferiscono ai file di definizione del formato degli scontrini PRINTER_E110.TXT e PRINTER_E120.TXT (vedi §4.8 a pag. 25), rispettivamente (non è comunque possibile usare i tasti numerici 6 e 7 per la selezione diretta di tali enquiries); infine, associa il tasto 0 alla digitazione manuale di causali numeriche "libere":

FKEY.TXT	*ENQUIRY.TXT*
[Functions]	*10,Riepilogo Presenze*
F1=D1	*110,Totale Accessi Odierni*
F5=D0	*120,Totale Accessi Mensili*
F2=R11	
F3=R22	
F4=E1	
F6=E2	
F7=E3	
F0=N	

4.6 DIRECTION.TXT

Lista degli orari in corrispondenza dei quali, ogni giorno, il criterio di scelta della singola direzione visualizzata cambia automaticamente: questo file ha effetto solo se il parametro **DirMode** all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 30) è stato impostato al valore 3 (default 4).

Si può usare un numero qualsiasi di record, il cui formato è il seguente:

HHMM_D<CR><LF>

Dove:

HHMM

HH=ore, MM=minuti

D

Criterio di scelta della direzione:

0 → Direzione preimpostata USCITA (è possibile usare il tasto [<-]-> per commutare la direzione, ma solo temporaneamente: una volta effettuata la transazione (o comunque dopo 10 secondi di inattività) la direzione torna ad essere quella preimpostata

1 → Direzione preimpostata ENTRATA (e uso del tasto [<-]-> come nel caso precedente)

2 → Nessuna direzione preimpostata. La direzione non viene più cambiata automaticamente: allo scattare dell'orario rimane quella precedentemente impostata, ma diventa possibile commutarla permanentemente tramite il tasto [<-]-> (potrà cambiare soltanto alla successiva pressione dello stesso tasto). Si tratta in pratica del funzionamento di default del parametro **DirMode**=3, cioè quello che si avrebbe se il file DIRECTION.TXT non fosse presente.

<CR><LF>

2 caratteri ASCII terminatori che devono essere sempre presenti in coda ad ogni record, compreso l'ultimo (ne consegue che il file deve sempre terminare con una linea vuota, e che la dimensione del file deve sempre essere un multiplo di 8 byte, che è la lunghezza fissa di ciascun record).

Nota: quando si carica il file DIRECTION.TXT la direzione visualizzata cambia in base all'orario corrente ma non lo fa immediatamente, bensì solo al primo cambio di minuto. Il criterio di scelta della direzione segue una logica di tipo "circolare", cioè la direzione impostata fino allo scattare del primo cambio orario elencato nel file è sempre quella relativa all'ultimo cambio orario elencato nello stesso file: ne consegue che se il file ha un solo record non vi sarà mai un cambio del criterio di scelta della direzione.

4.7 READER1.TXT, READER2.TXT, EXTREADER.TXT

Lista dei comandi di configurazione per un eventuale lettore seriale TTL (tipicamente un modulo R&W Mifare o Legic) di "console" collegato, rispettivamente, al connettore molex primario contrassegnato come "READER 1" (vedi figura della scheda a pag.9), al connettore molex secondario contrassegnato come "FINGER BOX" (o "READER 2" nelle versioni hardware fino alla 005, vedi §3.9 a pag. 13) o alla morsettiera a vite estraibile contrassegnata come "EXTERNAL READER".

Si vedano i "**RFID2 / RFID3 13.56MHz Reader Manual**" / "**RF4 13.56MHz Reader Manual**" / "**RF5 13.56MHz / 125KHz Reader Manual**" / "**RF4 13,56MHz Legic Reader Manual**" per una lista dei comandi disponibili per i moduli R&W Mifare o Legic con chipset SM-4200.

In particolare, per quanto riguarda il modulo R&W Legic, le cose funzionano diversamente a seconda del modello utilizzato:

- **Lettori RFID2/3 con chipset SC-2560**: l'unico comando disponibile è quello per definire le impostazioni di "autoread" del lettore (solo nel caso in cui si voglia leggere il contenuto di un segmento dati personalizzato, cioè quando il par. **CardDecode**=37; per "solo UID" è sufficiente impostare il par. **CardDecode**=36, vedi anche ***Nota 2*** in basso), che ha il seguente formato:

  CR A *XXX…XXX YY PP LL D*

  Dove:

  XXX…XXX è lo *stamp* del segmento in cui leggere, in formato HEX (esadecimale)

  YY è la lunghezza dello *stamp* in bytes (per compatibilità *Prime*), da 1 a 12

  PP è la posizione iniziale in bytes del codice da estrarre

  LL è la lunghezza in bytes del codice da estrarre

  D definisce il formato di decodifica:

  A: ASCII (Advant 0x00) – Ciascun *nibble* viene convertito in un carattere ASCII '0'..'F' (utile se si vuole che le cifre esadecimali 'A'..'F' siano accettate all'interno del codice tessera)

  B: BCD (Advant 0x01) – Ciascun *nibble* viene convertito in una cifra BCD '0'..'9' (le cifre esadecimali 'A'..'F' vengono sostituite con degli zeri '0')

  D: 4 Bytes su 10 cifre (Advant 0x05) – Ogni gruppo di 4 bytes viene convertito in decimale su 10 cifre (standard per "solo UID")

  N: Nessuno – Ciascun *nibble* produce una cifra decimale del codice tessera (le cifre esadecimali 'A'..'F' non sono accettate e generano un messaggio "Tessera non valida")

  Nota 1: è anche possibile inviare più(*) comandi CR A… simultaneamente, scrivendoli in un'unica linea, separati fra loro dal carattere *pipe* '|'.

Nota 2: In realtà, anche quando si usa solo lo UID, potrebbe essere necessario cambiare l'ordine dei byte nello UID in modo da mantenere la compatibilità con un vecchio formato speciale (usato per le carte *Prime* con UID di soli 4 byte). In particolare, rispetto allo UID standard trasmesso, in questo formato speciale il 2° ed il 4° byte sono scambiati fra loro. Potete cambiare il formato dello UID trasmesso dal default a quello scambiato (e quindi tornare al default, se necessario) impostando il par. **CardDecode**=37 (moduli Legic R&W con "autoread" personalizzato) e quindi usando uno dei seguenti comandi:

CR A 0 0 0 0 LS Seleziona il formato "UID Left "con ordine dei byte scambiato (1-4-3-2)

CR A 0 0 0 0 L Seleziona il formato "UID Left "con ordine dei byte standard (1-2-3-4)

- **Lettori RFID4 con chipset SCM-4200**: il comando più comunemente utilizzato è quello per definire le impostazioni di "autoread" del lettore (nel caso in cui si voglia leggere il contenuto di un segmento dati personalizzato oppure lo UID con un allineamento diverso da quello standard, cioè quando il par. **CardDecode**=30; per il semplice UID in formato standard è sufficiente impostare il par. **CardDecode**=42), che ha il seguente formato:

  CR A *XXX…XXX YY PP LL <flags>***\r**

  Dove:

  XXX…XXX è lo *stamp* del segmento in cui leggere, specificato in formato HEX (esadecimale)

  YY è la lunghezza dello *stamp* in bytes (per compatibilità *Prime*), da 1 a 12

  PP è la posizione iniziale in bytes del codice da estrarre

  LL è la lunghezza in bytes del codice da estrarre

  <flags> è una stringa contenente uno o più caratteri, ciascuno dei quali definisce un'impostazione specifica:

  - **Conversione**:

  A: ASCII (Advant 0x00) – Ciascun *nibble* viene convertito in un carattere ASCII '0'..'F' (utile se si vuole che le cifre esadecimali 'A'..'F' siano accettate all'interno del codice tessera) – solo per segmenti dati

  B: BCD (Advant 0x01) – Ciascun *nibble* viene convertito in una cifra BCD '0'..'9' (le cifre esadecimali 'A'..'F' vengono sostituite con degli zeri '0') – solo per segmenti dati

  D: 4 Bytes su 10 cifre (Advant 0x05) – Ogni gruppo di 4 bytes viene convertito in decimale su 10 cifre (default)

  - **Allineamento (solo per UID)**:

  L: Allineamento a sinistra (default)

  R: Allineamento a destra

  S: UID con byte ordinati secondo il formato *Prime* 1°-4°-3°-2° (si applica solo a trasponder di tipo Legic *Prime*)

  - **Tipo di trasponder (opzionale)**:

  Z: Legge solo Legic *Prime*

  W: Legge solo ISO15693

  Y: Legge solo ISO14443A

  Esempi:

  CR A 00 0 0 0 D\r

  Per leggere lo UID del primo trasponder rilevato in formato decimale (configurazione di default)

  CR A 000102 3 0 4 B\r

  Per leggere i primi 4 byte del primo segmento trovato con uno stamp che inizia con "000102", in formato BCD

  CR A 00 0 0 0 DY\r

  Per leggere lo UID del primo trasponder ISO 14443A rilevato in formato decimale

  CR A 00 0 0 0 DR\r

  Per leggere lo UID del primo trasponder rilevato con allineamento a destra

Nota: è anche possibile inviare più(*) comandi CR A… simultaneamente, scrivendoli in un'unica linea, separati fra loro dal carattere '+' e inserendo la stringa "\r" solo in coda all'ultimo.

Se il lettore di "console" è stato impostato per comunicazione seriale con configurazione di "autoread" custom (parametro **CardDecode**=30 / 33 / 37 nella corrispondente sezione *[Reader1]*, *[Reader2]* o *[ExtReader]* del file PARAMETERS.TXT, vedi a pag. 42), al primo cambio di parametro, o al primo riavvio, i comandi contenuti in questo file vengono mandati al lettore, quindi il file stesso viene cancellato. ***Nota***: se si utilizza un modulo R&W Mifare o Legic, prima di inviare i comandi di configurazione contenuti nel file viene automaticamente effettuato un reset del modulo al default di fabbrica.

Le risposte dell'eventuale modulo Mifare R&W ai comandi vengono scritte nel file LOG.TXT ma solo se il parametro **LogLevel** (vedi a pag. 51) è impostato almeno come "dettagliato" (valori 0 o 1, il default è 2).

Nota: è anche possibile inviare comandi manualmente e singolarmente ad un eventuale modulo R&W Mifare o Legic, mediante la corrispondente sezione **Reader 1**, **Reader 2** o **External Reader** del web server http del terminale. Questo è possibile solo se è stato precedentemente impostato il parametro **CardDecode**=30 / 32 / 33 / 37 / 42 / 43 nel file PARAMETERS.TXT oppure (il che è equivalente) sia stata selezionata una delle corrispondenti decodifiche di tipo "**Serial Reader**" dal menu a tendina "Card Decode" e applicata la modifica col pulsante "Save" nella medesima sezione del web server HTTP. In queste condizioni compare una casella di testo "Command" che in tutti gli altri casi non è presente, dove è possibile digitare il comando (nota: qualora sia necessario inserire un carattere terminatore *<CR>* occorre scrivere la stringa "\r" al suo posto; nel caso del modulo Legic R&W, per separare eventuali comandi multipli(*) CR A… è possibile usare il carattere *pipe* '|') che poi viene inviato col pulsante "Send command".

(*)Nota: ciascun comando CR A… inviato singolarmente cancella le precedenti impostazioni di "autoread" e imposta un singolo *search index*, ma se inviate più comandi CR A… tutti in una volta, allora potete impostare dei *search index* multipli.

4.8 PRINTER.TXT, PRINTER_R*CC..CC*.TXT E PRINTER_ENNN.TXT

Se è stata collegata una stampante e ne è stata abilitata la gestione, come descritto al §3.8 a pag. 13, è possibile stampare uno scontrino in seguito ad ogni transazione valida di rilevazione presenze o controllo accessi: lo scontrino ha un formato definito dal contenuto del file PRINTER.TXT, che può contenere stringhe di testo fisse e sequenze di caratteri speciali che vengono intercettate e sostituite con i dati relativi a ciascuna transazione al momento della stampa. E' anche possibile effettuare la stampa di scontrini con formati diversi in seguito ad una transazione valida di rilevazione presenze con selezione di una determinata causale, o in seguito alla semplice selezione di una *enquiry* locale (vedi §4.4 a pag. 19 e 4.5 a pag. 20): per ciascuna causale con codice *cc..cc* può essere usato il relativo file PRINTER_R*cc..cc*.TXT, che segue le stesse regole del file PRINTER.TXT ma viene stampato al posto di esso; analogamente, per ciascuna *enquiry* con identificativo *nnn* (si ricordi che deve essere *nnn* >= 100 per le *enquiries* da gestire in locale) può essere usato il relativo file PRINTER_E*nnn*.TXT, che segue anch'esso le stesse regole del file PRINTER.TXT. ***Avvertenza***: nei nomi dei file di stampa, i codici delle causali e delle *enquiries* locali devono essere identici a quelli indicati nei rispettivi file di definizione REASONS.TXT e ENQUIRY.TXT: eventuali zeri '0' a sinistra del codice causale *cc..cc* o dell'identificativo della *enquiry nnn* devono sempre essere specificati. Se invece si usa AXREASON.TXT (vedi §5.10 a pag. 79), gli zeri di riempimento del campo codice causale (avente lunghezza fissa 10) non devono essere specificati nel nome del file di stampa PRINTER_R*cc..cc*.TXT. A tale proposito, vediamo quali sono le sequenze di caratteri speciali che possono essere usate all'interno del file:

{*DD***}** produce la stampa del carattere il cui codice ASCII decimale è *DD*. Ad esempio, {13}{10} causa l'invio di *<CR><LF>*

%d produce la stampa di una stringa con la data corrente nel formato *gg/mm/aaaa* oppure *mm/gg/aaaa*, a seconda del valore del parametro **MonthDay** nella sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 30)

%h produce la stampa di una stringa con l'ora corrente nel formato *hh:mm:ss* oppure *hh:mm*, a seconda del valore del parametro **SecondsShown** nella sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 30)

%c produce la stampa del codice personale appena letto, come memorizzato nel file TRANSACTIONS.TXT - *da usare solo in PRINTER.TXT o PRINTER_Rcc..cc.TXT*

%v produce la stampa di una stringa relativa alla direzione del transito appena effettuato (**Entrata** o **Uscita**, o le analoghe stringhe nella lingua attualmente impostata) - *da usare solo in PRINTER.TXT o PRINTER_Rcc..cc.TXT*

%r produce la stampa della descrizione della causale se ne è stata selezionata una (altrimenti il campo rimarrà vuoto) - *da usare solo in PRINTER.TXT o PRINTER_Rcc..cc.TXT*

%u produce la stampa del nome dell'utente corrispondente al codice personale appena letto, se è presente il file USERS.TXT (vedi §5.9 a pag. 77, altrimenti il campo rimarrà vuoto) - *da usare solo in PRINTER.TXT o PRINTER_Rcc..cc.TXT*

%(*pattern_personalizzato***)** produce la stampa di una stringa che può contenere un numero qualunque di identificatori di campo da sostituire con i dati correnti al momento della stampa (ad esempio data e ora) e il cui formato è lo stesso che viene anche usato per il parametro **CustomRecord** all'interno sella sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §7.2 a pag. 101)

%A produce la stampa del numero totale di transazioni valide effettuate da tutti gli utenti nel mese corrente, su 5 caratteri allineati a destra con riempimento di spazi a sinistra

%B produce la stampa del numero totale di transazioni valide effettuate da tutti gli utenti nel giorno corrente, su 5 caratteri allineati a destra con riempimento di spazi a sinistra

%C produce la stampa del numero totale di transazioni valide effettuate da tutti gli utenti nel mese precedente, su 5 caratteri allineati a destra con riempimento di spazi a sinistra

%D produce la stampa del numero totale di transazioni valide effettuate da tutti gli utenti nel giorno precedente, su 5 caratteri allineati a destra con riempimento di spazi a sinistra

4.9 CONTROLLO REMOTO DEI RELE' E DEGLI INGRESSI DIGITALI DA WEB SERVER HTTP

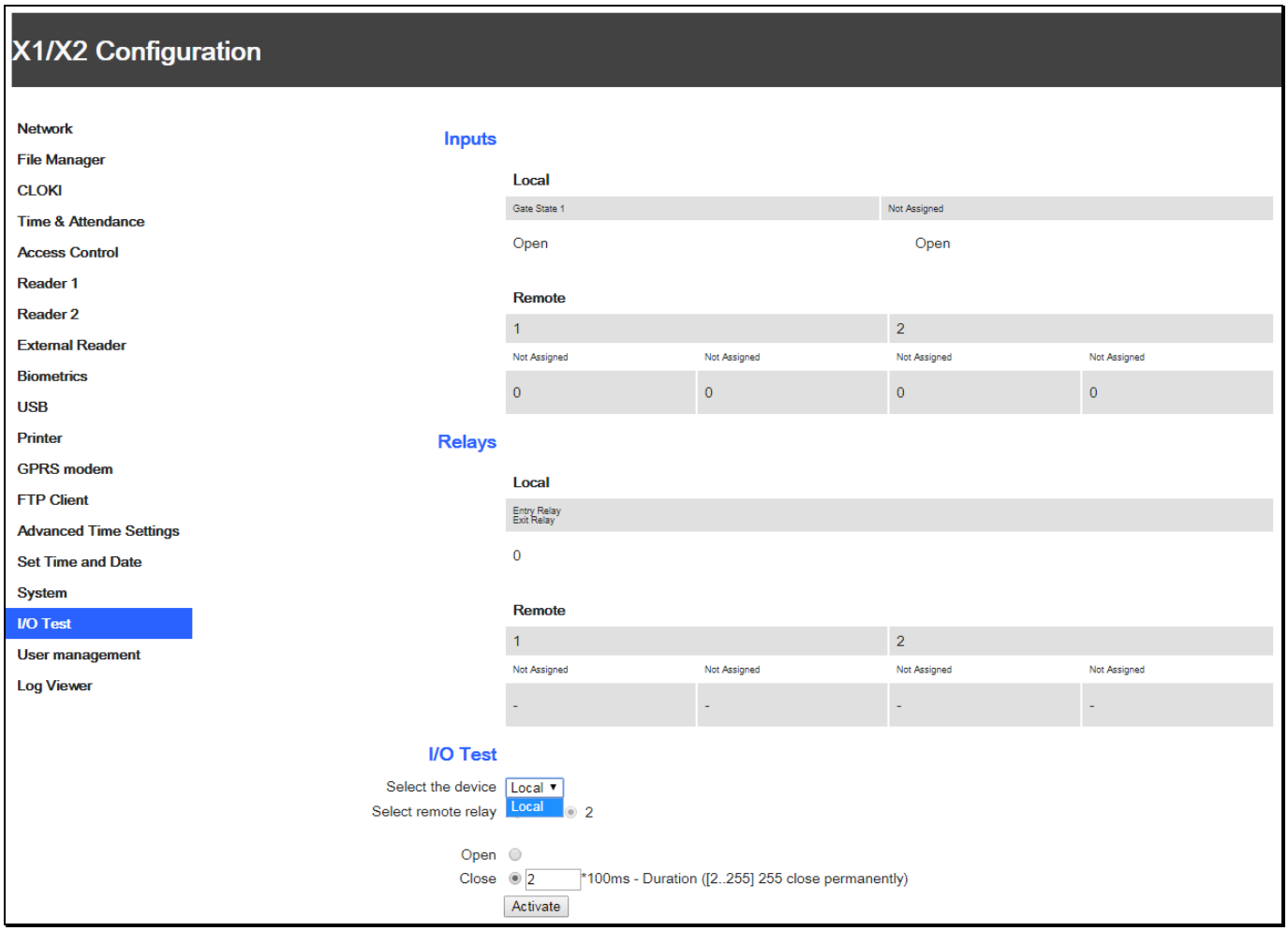
Mediante la sezione "**I/O Test**" del web server HTTP è anche possibile pilotare remotamente il relé interno di X1/X2 (quello indicato con il numero 1) e i 2 relé di ciascuna delle 2 eventuali schede di espansione 914 NeoMAX opzionali (indicati con i numeri 2 e 3 per quella con indirizzo RS485 '1' e con i numeri 4 e 5 per quella con indirizzo RS485 '2', anche se su ciascuna scheda NeoMAX compaiono come R1 e R2, rispettivamente).

Nota: il parametro **EnableNeoMaxI/O** nella sezione *[AccessControl]* del file PARAMETERS.TXT (vedi pag. 37) deve essere impostato a '1' (default) affinché le schede di espansione 914 NeoMAX vengano correttamente rilevate.

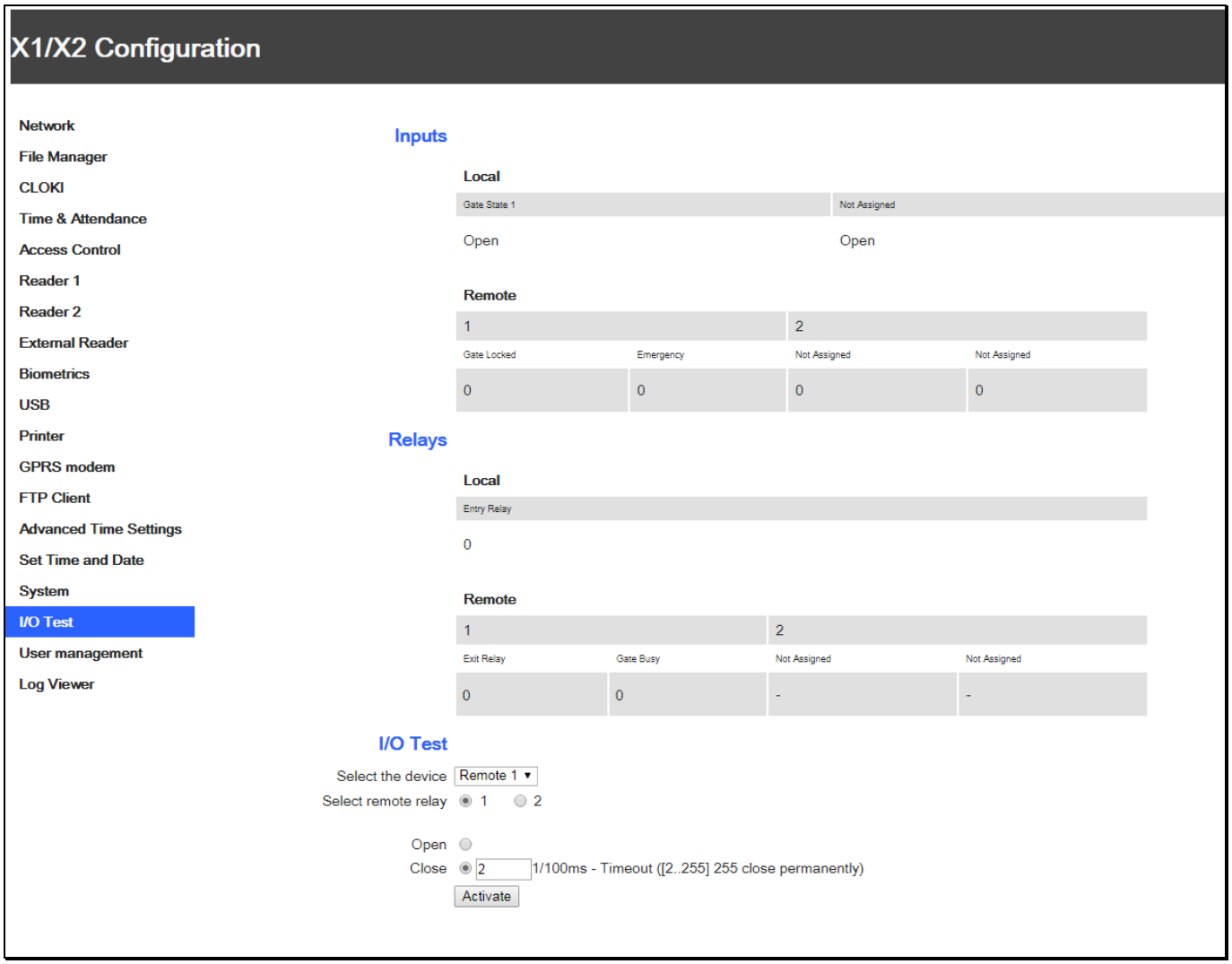
Nella pagina è anche mostrato lo stato corrente dei relé (il valore 0 significa relé non attivato), oltre allo stato corrente degli ingressi digitali interni di X1/X2 (quelli indicati con i numeri 1 e 2; i loro 2 possibili stati sono "Open", ovvero circuito aperto, e "Closed", ovvero corto circuito) e dei 2 ingressi digitali di ciascuna delle 2 eventuali schede di espansione 914 NeoMAX opzionali (indicati con i numeri 3 e 4 per quella con indirizzo 1 e con i numeri 5 e 6 per quella con indirizzo 2, anche se su ciascuna scheda NeoMAX compaiono come I1 e I2, rispettivamente; i loro 2 possibili stati sono 0, che significa circuito aperto, e 1). Analogamente, il valore 0 significa input non attivato. Tutti gli stati correnti visualizzati si riferiscono al momento del caricamento della pagina stessa: usate il pulsante "*aggiorna*" del browser, o "F5" sulla tastiera, per visualizzare i dati aggiornati).

Nell'esempio qui sotto le schede di espansione 914 NeoMAX non sono collegate, pertanto non compaiono gli stati relativi ai relé 2, 3, 4 e 5 né quelli relativi agli input 3, 4, 5 e 6 (gli indici '1' e '2' mostrati nella sezione "Remote" non sono stati, bensì gli indirizzi RS485 fissi che devono essere impostati tramite i *DIP switch* sui 914 NeoMAX affinché vengano rilevati), inoltre nella lista dei dispositivi da attivare compare solo la voce "Local", associata al relé 1 interno. Si può anche notare come tutti i gli

input e i relé abbiano come descrizione “Not Assigned”, ad eccezione dell’input 1 interno che è associato all’eventuale stato del varco, e del relé 1 interno che è associato all’eventuale apertura del varco sia per le transazioni in entrata che per quelle in uscita: questo è l’effetto della configurazione di default, in cui tutti i parametri che assegnano gli input per la gestione avanzata del varco (descritti al §6.3 a pag. 90) hanno il valore “0” tranne **GateSensor1** che vale appunto “1”, e tutti quelli che assegnano le uscite relé (descritti al §6.4 a pag. 92) hanno il valore “0” tranne **EntryRelay** e **ExitRelay** che valgono appunto “1”.



Nell’ulteriore esempio qui sotto, invece, una scheda di espansione 914 NeoMAX con indirizzo RS485 ‘1’ è collegata. Inoltre, è stato definito un varco controllato del tipo doppia porta impostando il par. **GateType**=“3” (vedi §6.1 a pag. 88), per cui gli input 1 e 2 vengono automaticamente assegnati allo stato della prima e della seconda porta (sempre che non vengano volutamente impostati valori diversi per i parametri **GateSensor1** e **GateSensor2**); è stato assegnato l’input 3 ad un pulsante per il blocco del varco impostando il par. **GateLocked**=”3”; è stato assegnato l’input 4 ad un pulsante di emergenza per lo sblocco continuo del varco impostando il par. **Emergency**=”4”; è stato assegnato il relé 2 all’apertura della seconda porta impostando il par. **ExitRelay**=”2”; è stato assegnato il relé 3 all’attivazione di una luce o un segnalatore acustico per segnalare dall’altra parte del varco la situazione di varco occupato impostando il par. **GateBusy**=”3”.



Dopo avere selezionato un dispositivo remoto si abilitano i *radio button* mediante i quali è possibile attivare o disattivare il relé corrispondente, mentre per il dispositivo "Local" viene comunque considerato l'unico relé presente a bordo di X1/X2. ***Nota***: Le voci "Close" e "Open" si riferiscono all'uscita normalmente aperta (**NO**), ma sia su X1/X2 che sui 914 NeoMAX sono presenti anche i contatti normalmente chiusi (**NC**): se si utilizzano questi contatti le voci visualizzate vanno interpretate al contrario. Il pulsante "Activate" consente in realtà di mandare il comando, sia per l'attivazione che per la disattivazione. E' possibile scegliere un valore finito del tempo di attivazione espresso in decimi di secondo, da un minimo di 2 (0.2s) ad un massimo di 254 (25.4s), oltre all'attivazione indefinita che si ottiene col valore 255. La disattivazione è immediata e ha effetto solo se il relé si trova in stato attivo (1).

4.10 IMPOSTAZIONE PARAMETRI

Metodo con file di testo

Al primo riavvio del terminale dopo la formattazione della micro-SD da PC, oppure in seguito all'utilizzo dell'opzione "**Reset default parameters**" dalla pagina "**System**" del web server HTTP di X1/X2 , un nuovo file di testo ASCII chiamato **PARAMETERS.TXT** viene creato automaticamente, con tutti i valori di default dei parametri (inclusa la configurazione IP).

Usando invece l'opzione "**Format SD Card**" dalla pagina "**System**" del web server HTTP di X1/X2, tutti gli altri file vengono cancellati, mentre il file PARAMETERS.TXT viene ricreato con tutti i valori di default ad eccezione della configurazione IP che

viene mantenuta, così come l'eventuale chiave di attivazione già inserita. La stessa cosa succede anche nel caso in cui il file PARAMETERS.TXT attualmente in uso venga cancellato.

Per cambiare la configurazione dei parametri, dovete solo caricare un nuovo PARAMETERS.TXT (deve contenere solo caratteri stampabili; il terminatore di linea deve essere CR+LF).

PARAMETERS.TXT ha la tipica struttura dei file .INI, con diverse sezioni come ad esempio *[Ethernet]* per le impostazioni Ethernet.

Le sezioni sono:

*[TimeAttendance], [AccessControl], [Reader1], [Reader2], [ExtReader], [Biometric], [System], [TimeSettings], [Ethernet], [GPRS], [FtpClient], [USB], [Printer], [Aperio]**

Nota: la sezione marcata con un asterisco (*) è presente solamente nella versione speciale di firmware Aperio (vedi §18 a pag. 165).

In ogni sezione, ciascuna linea (non fanno differenza caratteri minuscoli o maiuscoli) si riferisce ad un singolo parametro:

<nome_parametro>=<value>

All'avvio, e periodicamente in stato di inattività (non durante una transazione utente), X1 e X2 controllano questo file. Se viene trovato un file PARAMETERS.TXT con data più recente, allora i parametri specificati nel nuovo file vengono impostati, lasciando (o riportando) al default tutti quelli non ivi specificati.

Per cambiare solo alcuni parametri senza dover caricare un file PARAMETERS.TXT completo e senza il rischio di riportare al default eventuali altri parametri già cambiati in precedenza, è possibile invece usare un file con identica struttura chiamato UPDATECONF.TXT. In questo caso il file viene automaticamente rimosso dal terminale subito dopo avere effettuato le impostazioni relative ai parametri in esso contenuti.

Esempio di UPDATECONF.TXT che imposta un indirizzo IP statico:

; Ethernet section
[Ethernet]
DHCP=0
Ipaddress="192.168.1.240"

Metodo HTTP

Potete collegarvi alla pagina iniziale del web server del terminale (http://*<terminal_IP_Address>*), dove tutti i parametri possono essere consultati e modificati (in diverse sottopagine, che si possono selezionare mediante i link sul lato sinistro).

Per impostare i parametri è anche possibile usare un programma client HTTP che invia opportuni comandi in risposta ai messaggi "*Keep Alive*" ricevuti dal terminale. Si veda il §12.3 a pag. 140 per approfondire il concetto di messaggio "*Keep Alive*", ed il §12.4 a pag. 141 per sapere quali devono essere il formato della risposta del client HTTP ed i comandi di consultazione / impostazione del valore dei parametri.

4.11 LISTA DEI PARAMETRI

SEZIONE [TIMEATTENDANCE]

int **SecondsShown**

1: mostra i secondi sul display nella schermata principale (default)

0: i secondi non vengono mostrati

int **AmPm**

0: mostra l'orario nel formato 24 ore (default)

1: mostra l'orario nel formato am/pm

int **MonthDay**

0: mostra la data nel formato giorno/mese (default)

1: mostra la data nel formato mese/giorno

int **DateSeparator**

Indice decimale del carattere ASCII usato come separatore per la data. Il default è 47 → '/'

Nota: impostandolo dal menu del web server è possibile usare direttamente il carattere ASCII.

int **DirMode**

Direzioni di lettura del badge (default 4)

0: Solo Uscita

3: Entrata (In) o Uscita (Out), commutabili mediante il tasto [<-]-> (la direzione non viene più cambiata fino alla successiva pressione del tasto). Se è presente il file DIRECTION.TXT (vedi §4.6 a pag. 22), la direzione viene preimpostata e cambia automaticamente in base all'orario corrente e al contenuto del file. E' ancora possibile usare il tasto [<-]-> per commutare la direzione, ma solo temporaneamente: una volta effettuata la transazione (o comunque dopo 10 secondi di inattività) la direzione torna ad essere quella preimpostata in base all'orario corrente e al contenuto del file.

4: Out -> <- In (default)

5: In-> <- Out

6: Solo Entrata

int **BeepOk**

Imposta il tipo di suono emesso in caso di transazione valida:

0: Nessun suono

1..9: Numero di brevi "beep" monotonali emessi in sequenza

100 (Default): Suono politonale di default per le transazioni valide

int **BeepError**

Imposta il tipo di suono emesso in caso di transazione non valida:

0: Nessun suono

1..9: Numero di brevi "beep" monotonali emessi in sequenza

99 (Default): Suono politonale di default per le transazioni non valide

string **CompanyName**

Messaggio mostrato nella parte bassa dello schermo. Con **DirMode** impostato a 0 o 6, la lunghezza massima è di 21 caratteri, mentre con **DirMode** impostato a 4 o 5 la lunghezza massima è 12 caratteri.

Nota: con **DirMode** impostato a 3 la stringa **CompanyName** non può essere visualizzata.

int **ShowCode**

Tempo di visualizzazione del codice letto

0 → il codice della carta non viene mostrato sul display

Default 2 (secondi)

Max 99 (secondi)

int **RejectShorter**

1: se la lunghezza totale del codice contenuto nella carta è inferiore a **CardCodeLength**+**CardCodeBegin**, allora la carta viene rifiutata (default)

0: codici più corti vengono accettati comunque, e il codice utente viene riempito con zeri a sinistra

int **AllowTypeCode**

0: non è possibile digitare manualmente i codici utente nello stato di attesa transazioni (default)

1: i codici digitati manualmente vengono accettati (solo sui modelli X2 con tastiera numerica)

int **DisableReviewTA**

0 (default): la funzionalità di revisione dati di presenza (vedi §10.7 a pag. 118) è abilitata
1: disabilita la funzionalità di revisione dati di presenza

int **ReviewDaysTA**

Numero dei giorni precedenti al giorno corrente per i quali è possibile visualizzare le transazioni effettuate nella funzione di revisione dati di presenza (vedi §10.7 a pag. 118), procedendo a ritroso con il tasto "freccia su" (▲) a partire dall'ultima transazione effettuata. Default: 30. Esempio: impostando questo parametro al valore '1', verranno visualizzate solo le timbrature di oggi e di ieri, mentre con '0' verranno visualizzate solo le timbrature di oggi.

int **DisableTypeCodeReviewTA**

Ha effetto solo se i parametri **AllowTypeCode**=1 e **DisableReviewTA**=0. Per default, quando è possibile effettuare delle transazioni digitando i codici manualmente, è anche possibile accedere alla revisione dati di presenza (vedi §10.7 a pag. 118) digitando manualmente il codice di cui si vogliono visualizzare le transazioni precedentemente effettuate. Se si desidera disabilitare questa opzione, ad esempio per motivi di privacy, si può usare questo parametro:
0 (default): si può accedere alla revisione dati anche digitando manualmente il codice
1: non si può accedere alla revisione dati digitando manualmente il codice

int **Offline**

Determina le regole di validazione delle transazioni:

0 → ignora tutte le transazioni (nessuna registrazione in locale né trasmissione online)

1 → modalità offline: le transazioni vengono immediatamente validate e registrate in locale nel file TRANSACTIONS.txt. ***Nota***: se **MasterUrl** è impostato, le transazioni vengono comunque inviate all'host in tempo reale, ma solo per notifica e in modalità *batch*

2 → modalità online: le transazioni vengono inviate online in HTTP al **MasterUrl**. ***Nota***: le transazioni vengono anche registrate in locale nel file TRANSACTIONS.TXT una volta ricevuta la risposta dell'host, a meno che la risposta contenga il campo "**save=0**". Nel caso in cui l'host non risponda entro il timeout definito dal parametro **ConnTimeout** nella sezione [Ethernet] (valore di default 5 secondi), allora viene mostrato il messaggio di errore "**Server non connesso**" e non succede nulla. Le transazioni seguenti vengono immediatamente rifiutate, fino a quando il server non torna in linea (vedi §12.5 a pag. 144)

3 (default) → modalità semi-online: le transazioni vengono inviate online in HTTP al **MasterUrl**, e registrate in locale nel file TRANSACTIONS.TXT una volta ricevuta la risposta dell'host, a meno che la risposta contenga il campo "**save=0**". Nel caso in cui l'host non risponda entro il timeout definito dal parametro **ConnTimeout** (nella sezione [Ethernet], valore di default 5 secondi), allora le transazioni vengono validate e registrate in locale nel file TRANSACTIONS.txt. Le transazioni seguenti vengono immediatamente validate e registrate in locale, fino a quando il server non torna in linea (vedi §12.5 a pag. 144)

int **MaxPendingRecord**

Parametro che ha lo scopo di limitare la dimensione che può raggiungere il file riservato **btransactions.loc** (che contiene, in un formato fisso, tutte le ultime transazioni effettuate e tiene traccia anche di quelle che sono ancora "pendenti", cioè quelle non ancora trasmesse e/o non confermate dal server), in base alle regole descritte in dettaglio al §7 a pag. 96. Questo file riservato viene utilizzato anche come riferimento per la funzione di revisione dei dati di presenza locale, e rimane disponbile anche nel caso in cui il file **TRANSACTIONS.TXT** sia già stato scaricato e poi eliminato; il comportamento del terminale dipende dal valore del parametro **DeleteOld** in questa stessa sezione (vedi più sotto). Default: 12.000.

int **RepeatTimeOut**

Dopo una transazione, lo stesso codice tessera non viene più accettato per il tempo specificato

0 → Ripetute letture dello stesso codice vengono sempre accettate (default)

Max 99 (secondi)

int **DeleteOld**

Determina il comportamento del terminale quando viene effettuata una nuova transazione dopo che il numero totale delle transazioni precedentemente registrate all'interno del file **btransactions.loc** ha raggiunto il numero specificato dal parametro **MaxPendingRecord** (vedi sopra).

0 (default): a) se all'interno del file **btransactions.loc** corrente il numero delle sole transazioni ancora "pendenti" ha raggiunto il numero specificato dal parametro **MaxPendingRecord**, il terminale controlla la presenza del file **TRANSACTIONS.TXT**, assumendo implicitamente che si stia effettuando una gestione puramente offline: se tale file è presente, per evitare di bloccare ogni nuova registrazione e al contempo evitare di perdere traccia delle transazioni che potrebbero non essere ancora state scaricate via FTP, la nuova transazione viene regolarmente registrata all'interno del **btransactions.loc** già presente, sforando quindi il limite specificato dal parametro **MaxPendingRecord**; la stessa cosa avverrà per le transazioni successive, fintanto che chi gestisce il terminale non avrà scaricato e cancellato via FTP il file TRANSACTIONS.TXT : solo a quel punto, alla successiva transazione effettuata avverrà la rinomina dei file, esattamente come nel caso **DeleteOld**=1 descritto nel seguito. Se invece il file TRANSACTIONS.TXT non viene mai cancellato, il numero di transazioni ancora "pendenti" continuerà a incrementarsi fino a raggiungere il doppio del numero specificato dal parametro **MaxPendingRecord**: solo a quel punto il terminale si rifiuta di registrare ogni nuova transazione, mostrando a video il messaggio di errore "Err. memoria piena";

b) in caso contrario, tutto procede come nel caso **DeleteOld**=1 descritto nel seguito;

1: a) se non vi sono transazioni ancora "pendenti", o il numero delle sole transazioni ancora "pendenti" ha raggiunto il numero specificato dal parametro **MaxPendingRecord**, il file **btransactions.loc** corrente viene sempre rinominato in "btransactions.0.loc", e la nuova transazione viene memorizzata all'interno di un nuovo btransactions.loc. Allo stesso tempo, l'attuale file "TRANSACTIONS.TXT" viene rinominato in "TRANSACTIONS.0.TXT", e la nuova transazione viene memorizzata all'interno di un nuovo TRANSACTIONS.TXT. Ogni volta che viene raggiunto nuovamente il limite, ciascun precedente file "btransactions.*n*.loc" e "TRANSACTIONS.*n*.TXT", se presente, viene rinominato rispettivamente in "btransactions.*(n+1)*.loc" e "TRANSACTION.*(n+1)*.TXT", e il file corrente viene sempre rinominato rispettivamente in "btransactions.0.loc" e "TRANSACTIONS.0.TXT". I più vecchi file di transazioni possono essere "btransactions.3.loc" e "TRANSACTIONS.3.TXT", quindi in caso di successivi superamenti del valore massimo tali file vengono automaticamente cancellati. Con questo metodo, qualsiasi transazione nel file TRANSACTIONS.*n*.TXT file è sempre contenuta nel corrispondente file btransactions.*n*.loc con lo stesso valore *n*, e il numero di records in btransactions.loc è sempre uguale o maggiore del numero di records in TRANSACTIONS.TXT. Quando i file vengono rinominati, viene anche azzerato il contatore delle timbrature HTTP "pendenti".

b) in caso contrario, la registrazione prosegue all'interno del file btransactions.loc corrente.

string **CustomRecord**

Definisce un formato personalizzato per le transazioni memorizzate nel file TRANSACTIONS.TXT (vedi §7.2 a pag. 101).

Default: vuoto (viene utilizzato il formato standard, vedi §7 a pag.96). Lunghezza max: 68 caratteri.

string **CustomEntry**

Stringa che viene inserita nel campo "direzione di passaggio" (vedi §7.2 a pag. 101) per ogni transazione in entrata nel caso in cui sia impostato un formato personalizzato per il file TRANSACTIONS.TXT (par. **CustomRecord** non vuoto e contenente l'identificatore di campo 'V').

Default: "1"

string **CustomExit**

Come il parametro **CustomEntry** ma relativamente alle transazioni in uscita.

Default: "0"

int **BeepOnCard**

Da usare a scopo di debug, per controllare il tempo di reazione dell'host in modalità online

0: non viene emesso un beep subito dopo l'avvenuta lettura di una carta (default)

1: un breve suono "beep" (lo stesso usato per la pressione di un tasto) viene emesso subito dopo l'avvenuta lettura di una carta

string **ScreenOk**

Definisce un messaggio personalizzato da mostrare, in caso di transazione valida, al posto del messaggio standard "**Entrata**/**Uscita:** *codice_personale*" (oppure "*nome_utente*|**Entrata**/**Uscita**", se il controllo degli accessi è stato attivato, vedi §5 a pag. 68, e sono stati caricati i file CARDS.TXT, §5.4 a pag. 71, e USERS.TXT, §5.9 a pag. 77). Default: vuoto (viene mostrato il messaggio standard). All'interno del messaggio personalizzato è possibile inserire dei campi variabili usando i seguenti identificatori (*tag*), che verranno sostituiti dai rispettivi valori attuali al momento della transazione:

%c codice personale *oppure*
nome utente (alle stesse condizioni descritte sopra)

%v direzione (**Entrata** o **Uscita**)

%d orario (nel formato fisso *hh*:*mm*:*ss*)

%r descrizione della causale (se ne è stata selezionata una, altrimenti il campo rimarrà vuoto)

%a messaggio di errore standard (se usato dentro il parametro **ScreenError**, altrimenti il campo dà luogo alla stringa fissa "**Accesso Consentito**")

Oltre ai campi variabili, è possibile inserire qualunque carattere fisso, inclusi i caratteri speciali o non stampabili, fra i quali chr(24) che viene usato come carattere di controllo per il posizionamento della stringa che segue, vedi nota successiva. Per farlo, potete usare uno dei seguenti *tag*:

%h*XX* viene sostituito dal carattere il cui codice ASCII esadecimale è *XX* (espresso su 2 cifre)

{*DD***}** viene sostituito dal carattere il cui codice ASCII decimale è *DD*

***Nota*:** se non diversamente specificato, il messaggio personalizzato viene mostrato nella parte inferiore del display (la visualizzazione di data e ora rimane inalterata) su 2 linee di 21 caratteri ciascuna (max 42 caratteri visualizzabili), a partire dalla prima posizione della penultima linea e con ritorno a capo automatico al raggiungimento del 21esimo carattere. E' anche possibile usare il carattere di controllo chr(24) seguito da una lettera maiuscola che può assumere determinati valori a seconda che si voglia centrare la stringa che segue o allinearla a destra, e anche altri caratteri speciali per il posizionamento del cursore o la cancellazione del display:

{24}R *oppure* **%h18R** senza modificare la posizione verticale del cursore, centra sul display la stringa che segue fino al successivo carattere di controllo

{24}Q *oppure* **%h18Q** senza modificare la posizione verticale del cursore, allinea a destra la stringa che segue fino al successivo carattere di controllo

{24}N *oppure* **%h18N** posiziona il cursore all'inizio dell'ultima linea

'**|**' (*pipe*) *oppure* **{124}** *oppure* **%h7C** posiziona il cursore all'inizio della linea successiva a quella corrente

{12} *oppure* **%h0C** cancella il display e posiziona il cursore all'inizio della prima linea

string **ScreenError**

Definisce un messaggio personalizzato da mostrare, in caso di transazione non valida, al posto del messaggio di errore standard, che dipende dal motivo del rifiuto. Default: vuoto (viene mostrato il messaggio standard). All'interno del messaggio personalizzato è possibile inserire dei campi variabili usando gli stessi identificatori visti per il parametro **ScreenOk**, che verranno sostituiti dai rispettivi valori attuali al momento della transazione.

int **HideTypedCode**

Determina se durante la digitazione manuale del codice personale nello stato di attesa transazioni (solo se abilitata impostando il parametro **AllowTypeCode**=1 in questa stessa sezione, vedi pag. 31, e solo sui modelli X2 con tastiera numerica) il codice inserito debba essere visualizzato in chiaro oppure mascherato con asterischi, per evitare che venga visto e successivamente utilizzato da altri utenti non autorizzati.

0 → la digitazione manuale del codice non è mascherata (default)

1 → la digitazione manuale del codice è mascherata con asterischi. ***Nota***: in questo caso il codice personale non viene neanche mostrato per conferma in caso di transazione accettata (come invece succede normalmente, vedi §10.3 a pag. 111), e neppure in caso di codice inserito mediante lettura di tessera o identificazione biometrica, a prescindere dal valore del parametro **AllowTypeCode**; se però il controllo degli accessi è stato attivato (vedi §5 a pag. 68), e sono stati caricati i file CARDS.TXT (§5.4 a pag. 71) e USERS.TXT (§5.9 a pag. 77), viene comunque mostrato il nome dell'utente relativo a quel codice.

int **MultiFormat**

Se impostato a 1, consente di utilizzare diversi tipi di codifica o diversi criteri di controllo del codice comune e di estrazione del codice personale per carte di formati diversi, anche se vengono lette mediante lo stesso lettore. In pratica, per ogni lettura effettuata su un determinato lettore, X1/X2 applica dapprima la decodifica definita dal parametro **CardDecode** contenuto nella sezione relativa al lettore in questione, quindi effettua il controllo del codice comune in base al valore dei parametri **FacilityCodeBegin** e **FacilityCode** ed infine estrae il codice personale in base al valore dei parametri **CardCodeBegin** e **CardCodeLength** sempre di quella sezione.Nel caso in cui la tessera non risulti valida a causa di un errato codice comune o di un codice personale troppo corto, il terminale prova automaticamente ad applicare il tipo di decodifica e i criteri di elaborazione del codice definiti nelle sezioni relative ai rimanenti due lettori disponibili, partendo da quella più prioritaria (scala delle priorità in ordine decrescente: Reader1, Reader2, ExternalReader). ***Attenzione***: se il parametro **CardDecode** in una delle restanti sezioni è impostato a "99" (cioè il lettore è disabilitato), il relativo tracciato tessera non viene comunque applicato. Solo se la tessera non risulta valida neppure secondo i criteri relativi alle altre due sezioni, allora la transazione viene rifiutata. In caso contrario, appena si verifica la situazione in cui la tessera risulta essere valida secondo i criteri della sezione attualmente presa in esame, il codice personale viene passato alla gestione successiva (ad esempio alla logica di controllo accessi, se abilitato) come se la lettura fosse stata effettuata sul lettore relativo a quella sezione, anche se in realtà proviene da un altro lettore.

0 → vengono applicati solo il tipo di decodifica e i criteri di elaborazione del codice definiti nella sezione relativa al lettore su cui viene effettuata la lettura (default)

1 → se necessario, possono essere applicati anche il tipo di decodifica e i criteri di elaborazione del codice definiti nelle sezioni relative ai rimanenti due lettori disponibili, partendo da quella più prioritaria.

Nota: affinché sia possibile effettuare la decodifica dei dati provenienti da un certo lettore, le decodifiche alternative definite per gli altri lettori devono almeno fare riferimento allo stesso tipo di interfaccia, all'interno delle seguenti tipologie: Clock&Data (valori di **CardDecode** 0, 1, 3..26, 78, 79), seriale TTL (valori 30..33, ma solo se **BaudRateReader**=57600), Wiegand (valori 51..64). I lettori di carte magnetiche in tripla traccia (valori 81..85) al momento non supportano decodiche alternative.

int **MandatoryFunction**

E' anche possibile configurare il terminale in modo che non sia più possibile effettuare una transazione senza prima avere effettuato una selezione diretta della direzione o di una causale di scelta rapida,

Se impostato a 1, rende obbligatoria la selezione diretta della direzione di passaggio (a prescindere dall'eventuale direzione preimpostata dal terminale) o di una causale a scelta rapida subito prima della lettura di un badge, da effettuare mediante un qualunque tasto numerico 1..9, 0 (solo sui modelli X2 con tastiera numerica), selezionabile in base al contenuto del file FKEY.TXT (vedi §4.5 a pag. 20: per la selezione diretta della direzione di passaggio dovete riservare due diversi tasti numerici, uno per l'ENTRATA e uno per l'USCITA); in alternativa, anche sui modelli X1 che non dispongono della tastiera numerica, è possibile effettuare la selezione della direzione o di una causale mediante il menu di selezione "ridotto" (vedi §4.5 a pag. 20). Non sarà più possibile effettuare una transazione senza prima avere effettuato la selezione diretta della direzione o di una causale secondo una delle due modalità sopra descritte.

0 → è possibile effettuare transazioni con le modalità standard di selezione della direzione (default)

1 → è possibile effettuare transazioni solo in seguito alla selezione diretta della direzione o di una causale a scelta rapida mediante tasti numerici (solo sui modelli X2 con tastiera numerica) o menu di selezione "ridotto"

int **EnableFastMenu**

Non usato.

int **Payload**

Se impostato ad un valore diverso da 0, è possibile aggiungere, in coda ad ogni record nel file TRANSACTIONS.TXT (supponendo che si utilizzi il formato standard, vedere §7 a pag. 96) e dopo un carattere di separazione "|" *pipe* o chr(124), un ulteriore campo "payload esteso" il cui contenuto si suppone venga scelto fra un elenco di possibili opzioni; attualmente, comunque, l'unica opzione disponibile per questo campo è l'intero codice letto a seguito della decodifica applicata alla tessera e prima dell'estrazione del codice personale (indicato anche come "RAW data"), corrispondente al valore 1 del parametro.

Se si utilizza un formato di TRANSACTIONS.TXT personalizzato, è possibile aggiungere il campo di payload esteso utilizzando il segnaposto dedicato (vedere §7.2 a pag. 96).

0 (default): campo payload esteso disabilitato

1: salva la lettura completa del badge (RAW data") come payload esteso

int **AllowTypeReason**

Solo sui modelli X2 con tastiera numerica, se impostato ad un valore diverso da 0 consente l'introduzione di causali numeriche "libere" (ovvero non predefinite in REASONS.TXT né in AXREASON.TXT) mediante la digitazione manuale del codice causale prima di effettuare la lettura di tessera. Il codice causale inserito può essere lungo a piacimento (come per le causali predefinite, da un minimo di 1 ad un massimo di 8 cifre): una volta terminata la digitazione è possibile confermarlo premendo il tasto ↵ (Enter) e quindi leggere una tessera valida per effettuare la timbratura, o anche saltare la conferma passando direttamente alla lettura della tessera.

0 (default) → la digitazione manuale delle causali numeriche "libere" non è consentita nella schermata principale di stand-by: è eventualmente possibile solo in seguito alla pressione di un particolare tasto numerico di scelta rapida (vedi file FKEY.TXT al §4.5 a pag. 20).

1 → quando si preme un tasto numerico nella schermata principale di stand-by, esso viene automaticamente interpretato come la prima cifra di una casuale numerica "libera": è quindi possibile proseguire con l'inserimento delle cifre successive, oppure confermare e/o passare alla lettura della tessera. ***Avvertenze***: 1) questa modalità di funzionamento è incompatibile con l'utilizzo del file FKEY.TXT (vedi §4.5 a pag. 20), che quindi in questo caso non può essere usato per effettuare una scelta rapida, ma esclusivamente per definire il menu di selezione "ridotto" accessibile premendo il tasto ↵ (Enter) nella schermata di stand-by; 2) questa modalità di funzionamento è incompatibile con la modalità "digitazione manuale del codice utente": se il parametro **AllowTypeCode** in questa stessa sezione è impostato a '1', esso è più prioritario e quindi non è comunque consentita la digitazione manuale delle causali numeriche "libere".

2 → la digitazione manuale delle causali numeriche "libere" è consentita, partendo dalla schermata principale di stand-by, solo dopo avere premuto il tasto "freccia giù" (▼), che quindi in questo caso

non può più essere utilizzato per mostrare il menu delle causali predefinite, se presenti. E' eventualmente possibile fare la stessa cosa anche in seguito alla pressione di un particolare tasto numerico di scelta rapida (vedi file FKEY.TXT al §4.5 a pag. 20).

Nota: se in fase di digitazione manuale delle causali numeriche "libere" viene inserito un codice che coincide con quello di una delle causali predefinite in REASONS.TXT o in AXREASON.TXT, la relativa descrizione non viene in ogni caso mostrata durante la timbratura, ma sarà visibile in fase di revisione dati (vedi §10.7 a pag. 118).

SEZIONE [ACCESSCONTROL]

int **Enabled**

Abilita la funzionalità di controllo degli accessi (vedi §5 a pag. 68)

0 → controllo accessi non abilitato (default)

1 → abilita il controllo accessi

int **RelayActivation**

Tempo di attivazione relé per le transazioni offline, in decimi di secondo - default 5 (1/2 sec), max 255.

Nota: ha sempre effetto, indipendentemente dal fatto che il controllo accessi sia abilitato oppure no, o che la gestione avanzata del varco sia abilitata oppure no.

0 → non fa nulla

1 → disattiva semplicemente il relé nel caso fosse già attivo

255 →attiva il relé indefinitamente, da utilizzare solo per serrature "senza memoria" e se la gestione avanzata del varco è attivata (vedi parametro **GateEnabled** in questa stessa sezione a pag. 37): in questo caso, infatti, la disattivazione del relé deve necessariamente essere pilotata dall'attivazione dell'input relativo allo "stato porta" (vedi parametro **GateType** in questa stessa sezione), il che avviene quando il varco risulta effettivamente essere aperto.

int **EntryRelay**

Relé da attivare per le transazioni in entrata. I valori ammessi sono i seguenti (ogni altro valore non ha effetto, cioè non viene attivato nessun relé):

1 → relé interno già disponibile su X1/X2 (default)

2, 3 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (§3.3 a pag. 9)

4, 5 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **ExitRelay**

Relé da attivare per le transazioni in uscita (analogo a **EntryRelay**, di cui può anche avere lo stesso valore).

int **DeniedRelay**

Relé da attivare per segnalare una transazione non valida. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → relé interno già disponibile su X1/X2

2, 3 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (§3.3 a pag. 9)

4, 5 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

Nota: ha effetto anche se il controllo accessi non è abilitato (parametro **Enabled**=0).

int **DeniedRelayTimeout**

Tempo di attivazione del relé che segnala una transazione non valida, in decimi di secondo - default 5 (1/2 sec), max 255.

0 → non fa nulla

1 → disattiva semplicemente il relé nel caso fosse già attivo

255 →attiva il relé indefinitamente

int **Indexing**

Abilità la funzionalità di indicizzazione per i codici tessera ed i codici utente contenuti rispettivamente nei file CARDS.TXT e USERS.TXT. In questo modo vengono creati i corrispondenti file indice binari (chiamati CARDS.IDX e USERS.IDX), i quali consentono di ottenere la validazione in tempi quasi immediati (per 10000 codici sono necessari meno di 100ms rispetto ai 5-6 secondi della modalità standard).

0 → indicizzazione disabilitata (default)

1 → indicizzazione effettuata solo su aggiunta / cancellazione / modifica di un singolo record, come accade sempre quando X1/X2 è gestito dal programma XAtlas, oppure in seguito alla ricezione di un comando HTTP RA / RD / RM (vedi §12.4 a pag. 141); in questo caso i file CARDS.TXT e USERS.TXT non devono essere caricati via FTP, poiché i record non verrebbero indicizzati

2 → indicizzazione effettuata in blocco (ed in background) in seguito alla modifica dei file CARDS.TXT e USERS.TXT via FTP

int **PinOnly**

Abilita la modalità "solo PIN" (solo sui modelli X2 con tastiera numerica, vedi §5.12 a pag. 81)

0 → modalità "solo PIN" disabilitata (default)

1 → modalità "solo PIN" abilitata

int **AskPin**

Abilita la richiesta del codice PIN per gli utenti registrati nel file USERS.TXT e aventi un campo PPPP diverso da "0000" (vedi §5.9 a pag. 77)

0 → richiesta PIN disabilitata (default)

1 → richiesta PIN abilitata

Int **FullTable**

Usato solo quando X1/X2 viene gestito dal programma Xatl@s: consente di scegliere se sia possibile effettuare l'accesso anche a tabelle caricate parzialmente oppure solo quando siano state caricate completamente.

0 → Accesso consentito anche a tabelle caricate parzialmente (default)

1 → Accesso consentito solo a tabelle caricate completamente

int **RecordInvalidAccess**

Abilita la registrazione nel file TRANSACTIONS.TXT di tutti i tentativi di accesso autorizzati e non autorizzati (inclusi quelli risultati non autorizzati secondo i criteri di controllo accessi). Le transazioni non autorizzate sono distinguibili da quelle valide poiché nel campo "CONTROLLI" viene registrato un valore diverso da '00' e il campo "ESITO" contiene il valore '1' invece di '0' (vedi §7 a pag. 96).

Nota: questa impostazione ha senso solo se il controllo accessi è stato attivato impostando a 1 il parametro **Enabled**.

Attenzione: non vengono comunque mai registrate le letture che hanno fallito i controlli preliminari relativi alla correttezza formale del codice letto (lunghezza e codice comune).

0 → vengono registrate solo le transazioni autorizzate (default)

1 → vengono registrati tutti i tentativi di accesso autorizzati e non autorizzati

int **EnableNeoMaxI/O**

Consente di disabilitare il *polling* automatico (ovvero l'interrogazione periodica) delle eventuali schede di espansione 914 NeoMAX collegate sulla linea RS485, nel caso in cui non si abbia intenzione di usarle: questo consente di ottimizzare le risorse e migliorare le prestazioni delle comunicazioni TCP di circa il 15%.

0 → disattiva il polling delle eventuali schede di espansione 914 NeoMAX

1 → polling delle schede 914 NeoMAX abilitato (default)

int **GateEnabled**

Abilita la funzionalità di gestione avanzata di un varco (vedi §6 a pag. 88)

0 → gestione avanzata del varco non abilitata (default)

1 → abilita la gestione avanzata del varco

int **GateType**

Definisce il tipo di varco da gestire:

0 → varco non controllato (default). Se la gestione avanzata del varco è attivata (par. **GateEnabled**=1) ma il varco non è controllato, l'unica differenza rispetto al caso di gestione avanzata del varco non attivata è che il messaggio "*Keep Alive*" inviato periodicamente all'host (vedi §6.5 a pag. 93) contiene sempre il *server tag* "**gateStatus=H**" (cioè "stato normale")

1 → porta battente: in questo caso, se non diversamente specificato impostando il parametro **GateSensor1** ad un valore diverso da '1' e assegnando il valore '1' a uno degli altri parametri elencati al §6.3 a pag. 90, l'input IN1 (già disponibile su X1/X2) viene automaticamente assegnato allo stato della porta

2 → tornello: in questo caso, se non diversamente specificato impostando il parametro **GateSensor1** ad un valore diverso da '1' e assegnando il valore '1' a uno degli altri parametri elencati al §6.3 a pag. 90, l'input IN1 (già disponibile su X1/X2) viene automaticamente assegnato allo stato del tornello

3 → doppia porta o bussola di sicurezza: in questo caso, se non diversamente specificato impostando il parametro **GateSensor1** ad un valore diverso da '1' ed il parametro **GateSensor2** ad un valore diverso da '2', e assegnando i valori '1' e '2' ad altri due parametri fra quelli elencati al §6.3 a pag. 90, l'input IN1 (già disponibile su X1/X2) viene automaticamente assegnato allo stato della prima porta, e l'input IN2 (anch'esso già disponibilesu X1/X2) allo stato della seconda porta

4 → centrale allarme: valore riservato alla gestione di X1/X2 da parte del programma Xatl@s

Vedi anche il parametro **GateState1** a pag. 40 per definire lo stato a riposo della porta/tornello e il parametro **GateState2** per definire lo stato a riposo della seconda porta (solo nel caso di doppia porta/bussola).

int **TimeOutOpen**

Definisce il tempo massimo (in decimi di secondo) entro il quale deve essere fisicamente aperto il varco per iniziare l'attraversamento dopo lo sblocco in seguito ad una transazione valida. Default: 50 (5 secondi).

int **TimeOutOpenExtended**

Come il precedente parametro **TimeOutOpen**, ma valido solo per gli utenti a cui è consentito un tempo più lungo per effettuare l'accesso (vedi flag 'T' nei record del file USERS.TXT, §5.9 a pag. 77). Default: 100 (10 secondi).

int **TimeOutClose**

Definisce il tempo massimo (in decimi di secondo) entro il quale deve essere fisicamente richiuso il varco a partire dal momento in cui viene aperto in seguito ad una transazione valida. Default: 50 (5 secondi).

int **TimeOutCloseExtended**

Come il precedente parametro **TimeOutClose**, ma valido solo per gli utenti a cui è consentito un tempo più lungo per effettuare l'accesso (vedi flag 'T' nei record del file USERS.TXT, §5.9 a pag. 77). Default: 100 (10 secondi).

int **ManualUnlockIN**

Input usato per gestire un pulsante di sblocco manuale del varco per una singola entrata. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → input IN1, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor1**, come per default)

2 → input IN2, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor2**, come per default in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **ManualUnlockOUT**

Input usato per gestire un pulsante di sblocco manuale varco per una singola uscita. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → input IN1, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor1**, come per default)

2 → input IN2, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor2**, come per default in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **Emergency**

Input usato per gestire un pulsante di sblocco manuale continuo del varco in caso di emergenza. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → input IN1, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor1**, come per default)

2 → input IN2, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor2**, come per default in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **EmergencyRelay**

Relé da attivare per segnalare la situazione di emergenza generata dall'attivazione manuale dell'input associato al precedente parametro **Emergency**. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → relé interno già disponibile su X1/X2

2, 3 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (§3.3 a pag. 9)

4, 5 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **GateLocked**

Input usato per gestire un pulsante di blocco manuale continuo del varco. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → input IN1, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor1**, come per default)

2 → input IN2, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor2**, come per default in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **GateLockedRelay**

Relé da attivare per segnalare la situazione di blocco del varco generata dall'attivazione manuale dell'input associato al precedente parametro **GateLocked**. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → relé interno già disponibile su X1/X2

2, 3 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (§3.3 a pag. 9)

4, 5 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **GateAlert**

Relé da attivare per segnalare la situazione di allarme generata da un cambiamento di stato dell'input IN1 (sempre associato allo stato della porta o del tornello, vedi parametro **GateType** a pag. 38, o dell'input IN2 in caso di varco di tipo doppia porta o bussola di sicurezza) quando il varco è chiuso (varco forzato), o dall'attivazione dell'input associato al parametro **TurnstileAlert** (effrazione, solo in caso di varco di tipo tornello, vedi a pag. 41). I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → relé interno già disponibile su X1/X2

2, 3 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (§3.3 a pag. 9)

4, 5 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **GateTransitOk**

Relé da attivare per segnalare la situazione di varco sbloccato in seguito ad una transazione valida o all'attivazione degli input associati ai parametri **ManualUnlockIN** e **ManualUnlockOUT** (sblocco manuale da pulsante per singola entrata o uscita, vedi a pag. 38). I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → relé interno già disponibile su X1/X2

2, 3 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (§3.3 a pag. 9)

4, 5 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **InterLocked**

Input usato per bloccare il terminale finché il varco è impegnato poiché è in corso un transito in direzione opposta. Utile nel caso in cui vengano usati 2 terminali diversi sui lati opposti di uno stesso varco. Questo input deve essere collegato all'altro terminale sull'uscita relé definita dal rispettivo parametro **GateBusy** descritto qui sotto. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → input IN1, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor1**, come per default)

2 → input IN2, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor2**, come per default in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **GateBusy**

Relé da attivare per segnalare che il varco è impegnato. Utile nel caso in cui vengano usati 2 terminali diversi sui lati opposti di uno stesso varco. Questa uscita relé deve essere collegata all'altro terminale sull'input definito dal rispettivo parametro **InterLocked** sopra descritto, per segnalargli che non è possibile effettuare transiti. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → relé interno già disponibile su X1/X2

2, 3 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (§3.3 a pag. 9)

4, 5 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **GateSensor1**

Input usato per controllare lo stato del varco (aperto/chiuso). Default: 1. Lo stato a riposo dell'input che è stato assegnato a questo scopo è definito dal parametro **GateState1**.

0 → non gestito (valore non ammesso in caso di varco controllato)

1 → input IN1, già disponibile su X1/X2 (default)

2 → input IN2, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor2**, come per default in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **GateSensor2**

Input usato per controllare lo stato della seconda porta (aperta/chiusa). Default: 0. Default in caso di impostazione varco di tipo doppia porta o bussola di sicurezza: 2. Lo stato a riposo dell'input che è stato assegnato a questo scopo è definito dal parametro **GateState2**.

0 → non gestito (default; valore non ammesso in caso di varco di tipo doppia porta o bussola di sicurezza)

1 → input IN1, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor1**, come per default)

2 → input IN2, già disponibile su X1/X2 (default in caso di impostazione varco di tipo doppia porta o bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **GateState1**

Definisce lo stato a riposo dell'input impostato dal parametro **GateSensor1** che segnala l'apertura / chiusura della porta o del tornello in caso di varco controllato (parametro **GateType** diverso da 0, vedi a pag. 38).

0 → Input aperto (non attivo) con varco chiuso

1 → Input cortocircuitato (attivo) con varco chiuso (default)

int **GateState2**

Definisce lo stato a riposo dell'input impostato dal parametro **GateSensor2** che segnala l'apertura / chiusura della seconda porta nel caso di varco del tipo doppia porta o bussola di sicurezza (parametro **GateType**=3, vedi a pag. 38).

0 → Input aperto (non attivo) con varco chiuso

1 → Input cortocircuitato (attivo) con varco chiuso (default)

int **ExternalNoTransit**

Ha effetto solo se è stato selezionato un varco del tipo tornello (parametro **GateType**=2, vedi a pag. 38): definisce l'input usato per ricevere una segnalazione di transito non avvenuto da una logica esterna, normalmente usata nei tornelli. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → input IN1, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor1**, come per default)

2 → input IN2, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor2**, come per default in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **TurnstileAlert**

Ha effetto solo se è stato selezionato un varco del tipo tornello (parametro **GateType**=2, vedi a pag. 38): definisce l'input usato per ricevere una segnalazione di effrazione da una logica esterna, normalmente usata nei tornelli. I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → input IN1, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor1**, come per default)

2 → input IN2, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor2**, come per default in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **SecurityBoothAuth**

Ha effetto solo se è stato selezionato un varco del tipo doppia porta o bussola di sicurezza (parametro **GateType**=3, vedi a pag. 38): definisce l'input usato per ricevere, da parte di una logica esterna normalmente usata nelle bussole, la segnalazione relativa al fatto che è possibile aprire la seconda porta. Questo parametro va usato solo se l'uscita della logica esterna è normalmente bassa (input aperto, cioè non attivo, a riposo). In caso contrario, occorre usare, in alternativa, il seguente parametro **SecurityBoothAuthDeny**.

0 → non gestito (default)

1 → input IN1, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor1**, come per default)

2 → input IN2, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor2**, come per default in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **SecurityBoothAuthDeny**

Ha effetto solo se è stato selezionato un varco del tipo doppia porta o bussola di sicurezza (parametro **GateType**=3, vedi a pag. 38): definisce l'input usato per ricevere, da parte di una logica esterna normalmente usata nelle bussole, la segnalazione relativa al fatto che non è ancora possibile aprire la seconda porta. Questo parametro va usato solo se l'uscita della logica esterna è normalmente alta (input cortocircuitato, cioè attivo, a riposo). In caso contrario, occorre usare, in alternativa, il precedente parametro **SecurityBoothAuth**.

0 → non gestito (default)

1 → input IN1, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor1**, come per default)

2 → input IN2, già disponibile su X1/X2 (se non già usato per il parametro **GateSensor2**, come per default in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **PendingAlarms**

Parametro riservato per la gestione di X1/X2 da parte del programma XAtl@s quando si imposta il tipo di varco "alarm manager" (default: 0).

SEZIONE [READER1]

int **CardDecode**

Selezione decodifica della carta:

0: lettore di carte magnetiche in traccia 2 (o altro lettore con uscita compatibile) *oppure*

RFID 125KHz TMC standard (14 cifre decimali) *oppure*

HID 37bit "Clock&Data" H10320 (8 cifre decimali)

1: RFID 125KHz Unique o "Nord-Europea" (13 cifre decimali)

2: carte barcode

3: RFID 125KHz Dating (4+1+8=13 cifre decimali)

4: RFID 125KHz Apice/Cronos (3+10=13 cifre decimali)

5: RFID 125KHz a gruppi (4+5+5=14 cifre decimali)

6: RFID 125KHz EM4102 su soli 4 bytes (11 cifre decimali)

7: RFID 125KHz Crosspoint (9 cifre decimali)

8: RFID 125KHz Zucchetti (13 cifre decimali)

9: RFID 125KHz Dating "4° nibble" (4+9=13 cifre decimali)

10: RFID 125KHz Kronotech (20 cifre decimali)

11: RFID 125KHz Byte (8 cifre decimali)

12: RFID 125KHz BCD (10 cifre decimali)

13: HID Clock&Data 26bit H10301 (8bit facility + 16bit user -> 3+5=8 cifre decimali)

14: HID Clock&Data 34bit H10306 (5+5=10 cifre decimali)

15:HID Clock&Data 37bit H10304 (16bit facility +19bit user -> 5+7=12 cifre decimali)

16: HID Clock&Data 37bit H10302 (11 cifre decimali)

17: HID Clock&Data 40bit formato Wiegand (4+5=9 cifre decimali)

18: HID Clock&Data 35bit Corporate 1000 (4+7=11 cifre decimali)

19: HID Clock&Data 32bit (13bit facility + 17bit user -> 4+6=10 cifre decimali)

20: HID Clock&Data 32bit (15bit facility + 15bit user -> 5+5=10 cifre decimali)

21: HID Clock&Data 36bit (17bit facility + 16bit user -> 6+5=11 cifre decimali)

22: HID Clock&Data 36bit (8bit facility + 24bit user -> 3+8=11 cifre decimali)

23: HID Clock&Data 36bit (12bit facility + 20bit user -> 5+6=11 cifre decimali)

24: HID Clock&Data 30bit (8bit facility + 20bit user -> 3+6=9 cifre decimali)

25: HID Clock&Data 37bit BCD (9 cifre decimali)

26: HID Clock&Data 35bit Corporate 1000 *oppure* 40bit formato Wiegand (40 cifre decimali in entrambi I casi)

27: HID Clock&Data 46bit (16bit facility + 28bit user -> 5+9=14 cifre decimali)

28: HID Clock&Data 48bit Corporate 1000 (7+7=14 cifre decimali)

30 (default): lettore RFID2/3/4 13,56MHz *oppure* RF5 125KHz + 13,56MHz seriale *oppure* RF4 Legic seriale con chipset SM-4200 – con "autoread" custom: il formato e la lunghezza del codice letto dipendono dal tipo di carta e dalla configurazione di "autoread" attualmente memorizzata nel lettore, che non viene cambiata dal terminale (il default è la lettura del codice UID a 10 o 20 cifre decimali). ***Attenzione***: per un corretto funzionamento, il successivo parametro **BaudrateReader** deve essere impostato al valore di default (57600).

31: lettore RFID HID iClass seriale TTL – decodifica del cosiddetto "Wiegand Data" che normalmente è il codice stampato sulle carte HID iClass (il quale è diverso dal codice UID leggibile anche con i lettori 13,56MHz standard). ***Attenzione***: per un corretto funzionamento, il successivo parametro **BaudrateReader** deve essere impostato al valore di default (57600).

32: lettore RFID2/3/4/5 seriale 13,56MHz – codifica custom TMC per carte Mifare R&W: il codice deve essere scritto nel blocco dati il cui indice è impostato dal parametro **MifareFirstBlock** nella sezione *[Biometric]* (vedi a pag. 49), a partire dall'offset 1 (quindi escluso il primo byte) per una lunghezza di 8 byte in formato packed BCD (2 cifre per byte), quindi in totale il codice può contenere fino a 16 cifre decimali significative, con eventuali zeri di riempimento a sinistra. **Nota**: alle cifre significative lette vengono comunque aggiunti tanti zeri di riempimento quanti sono necessari per raggiungere una lunghezza pari alla somma dei valori dei parametri **CardCodeBegin** e **CardCodeLength** relativi al lettore. ***Attenzione***: ogni eventuale configurazione di "autoread" preimpostata verrà automaticamente rimossa; per un corretto funzionamento, il successivo parametro **BaudrateReader** deve essere impostato al valore di default (57600).

33: lettore seriale generico: il formato e la lunghezza del codice letto dipendono esclusivamente dal tipo di lettore collegato, la cui baudrate può essere impostata a piacimento mediante il successivo parametro **BaudrateReader** (default 57600).

36: lettore RFID2/3 Legic seriale con chipset SC-2560 – solo lettura del codice UID a 10 o 20 cifre decimali. ***Attenzione***: ogni eventuale configurazione di "autoread" preimpostata verrà automaticamente rimossa; per un corretto funzionamento, si consiglia di impostare il successivo parametro **BaudrateReader** al valore 0 (autorilevamento della baudrate).

37: lettore RFID2/3 Legic seriale con chipset SC-2560 con "autoread" custom: il formato e la lunghezza del codice letto dipendono dal tipo di carta e dalla configurazione di "autoread" attualmente memorizzata nel lettore, che non viene cambiata dal terminale (il default è la lettura del codice UID a 10 o 20 cifre decimali). ***Attenzione***: per un corretto funzionamento, si consiglia di impostare il successivo parametro **BaudrateReader** al valore 0 (autorilevamento della baudrate).

38: lettore RFID2/3 Legic seriale con chipset SC-2560 Tag A – valore custom riservato. ***Attenzione***: ogni eventuale configurazione di "autoread" preimpostata verrà automaticamente rimossa.

41: lettore RFID2/3/4/5 seriale 13,56MHz – codifica Zucchetti e visualizzazione a display del nome utente contenuto nella carta Mifare (richiede chiave di attivazione, vedi §4.12 a pag. 65). ***Attenzione***: ogni eventuale configurazione di "autoread" preimpostata verrà automaticamente rimossa; per un corretto

funzionamento, il successivo parametro **BaudrateReader** deve essere impostato al valore di default (57600).

42: lettore RFID2/3/4/5 13,56MHz seriale *oppure* RF4 Legic seriale con chipset SM-4200 – solo lettura del codice UID a 10 o 20 cifre decimali. ***Attenzione***: ogni eventuale configurazione di "autoread" preimpostata verrà automaticamente rimossa; per un corretto funzionamento, il successivo parametro **BaudrateReader** deve essere impostato al valore di default (57600).

43: lettore RFID2/3/4/5 13,56MHz seriale – solo lettura del codice UID Reverse a 10 o 20 cifre decimali (i byte vengono letti in ordine inverso). ***Attenzione***: ogni eventuale configurazione di "autoread" preimpostata verrà automaticamente rimossa; per un corretto funzionamento, il successivo parametro **BaudrateReader** deve essere impostato al valore di default (57600).

47: lettore RFID HITAG seriale TTL – 10 cifre decimali. ***Attenzione***: per un corretto funzionamento, il successivo parametro **BaudrateReader** deve essere impostato al valore '9600', ed il parametro RepeatTimeOut nella sezione *[TimeAttendance]* (vedi pag. 32) deve essere impostato come minimo al valore '2' (o superiore).

51: Lettore generico con uscita Wiegand 26bit H10301 (8bit facility + 16bit user -> 3+5=8 cifre decimali)

52: Lettore generico con uscita Wiegand 34bit H10306 (5+5=10 cifre decimali)

53: Lettore generico con uscita Wiegand 37bit H10304 (16bit facility + 19bit user -> 5+7=12 cifre decimali)

54: Lettore generico con uscita Wiegand 37bit H10302 (11 cifre decimali)

55: Lettore generico con uscita Wiegand 40bit formato Wiegand (4+5=9 cifre decimali)

56: Lettore generico con uscita Wiegand 35bit Corporate 1000 (4+7=11 cifre decimali)

57: Lettore generico con uscita Wiegand 32bit (13bit facility + 17bit user -> 4+6=10 cifre decimali)

58: Lettore generico con uscita Wiegand 32bit (15bit facility + 15bit user -> 5+5=10 cifre decimali)

59: Lettore generico con uscita Wiegand 36bit (17bit facility + 16bit user -> 6+5=11 cifre decimali)

60: Lettore generico con uscita Wiegand 36bit (8bit facility + 24bit user -> 3+8=11 cifre decimali)

61: Lettore generico con uscita Wiegand 36bit (12bit facility + 20bit user -> 5+6=11 cifre decimali)

62: Lettore generico con uscita Wiegand 30bit (8bit facility + 20bit user -> 3+6=9 cifre decimali)

63: Lettore generico con uscita Wiegand 37bit BCD (9 cifre decimali)

64: Lettore generico con uscita Wiegand 35bit Corporate 1000 (40 cifre decimali)

65: Lettore generico con uscita Wiegand 46bit (16bit facility + 28bit user -> 5+9=14 cifre decimali)

66: Lettore generico con uscita Wiegand 32bit (32 bit user -> 10 cifre decimali)

67: Lettore generico con uscita Wiegand 64bit (16+16+32bit -> 5+5+10=20 cifre decimali)

68: Lettore generico con uscita Wiegand 52bit BCD (13 cifre decimali)

69: Lettore generico con uscita Wiegand 48bit Corporate 1000 (7+7=14 cifre decimali)

70: Lettore generico con uscita Wiegand 27bit (13bit facility + 14bit user -> 4+4=8 cifre decimali)

74: RFID 125kHz Octal (14 cifre ottali)

75: RFID 125kHz Byte 10 Nibbles (10 cifre decimali): funziona come per il valore 11, ma si applica a tutti e 5 i byte degli UID delle carte invece che ai soli 4 byte meno significativi

76: RFID 125KHz Microntel (10 cifre decimali)

77: RFID 125KHz Kronotech Mirrored (20 cifre decimali)

78: RFID 125KHz ASCII Hex (10 cifre esadecimali)

79: RFID 125KHz ASCII Hex Reverse (senza ribaltamento dei *nibble* - 10 cifre esadecimali)

81: lettore di carte magnetiche in tripla traccia – Emette l'intero contenuto, inclusi gli eventuali separatori di campo, della prima traccia codificata incontrata seguendo l'ordine Tk1->Tk2->Tk3

82: lettore di carte magnetiche in tripla traccia – Emette l'intero contenuto, inclusi gli eventuali separatori di campo, della prima traccia codificata incontrata seguendo l'ordine Tk3->Tk2->Tk1

83: lettore di carte magnetiche in tripla traccia – Emette l'intero contenuto, inclusi gli eventuali separatori di campo, della sola traccia 1 (se presente, altrimenti non emette nulla)

84: lettore di carte magnetiche in tripla traccia – Emette l'intero contenuto, inclusi gli eventuali separatori di campo, della sola traccia 2 (se presente, altrimenti non emette nulla)

85: lettore di carte magnetiche in tripla traccia – Emette l'intero contenuto, inclusi gli eventuali separatori di campo, della sola traccia 3 (se presente, altrimenti non emette nulla)

86: lettore di carte magnetiche a singola traccia modificato per la traccia 1

91: lettore seriale RFID2/3 TTL 13,56MHz – da usare esclusivamente con la versione speciale di firmware "Aperio", che consente di gestire le carte di un impianto di serrature wireless offline della serie Aperio (vedi §18 a pag. 165).

99: Lettore disabilitato.

int **BaudrateReader**

Significativo solo in caso di lettore seriale TTL (**CardDecode** = 30..38). Default 57600. Altri valori ammessi: 38400, 19200, 9600, 0 (autorilevamento della baudrate, solo per lettori Legic, ovvero **CardDecode** = 36..38).

int **CardLayoutLength**

Se impostato ad un valore diverso da 0, specifica la lunghezza che deve avere l'intero codice letto a seguito della decodifica applicata alla tessera e prima dell'estrazione del codice personale (noto anche come "RAW data"): anche se l'estrazione del codice personale (in base al valore dei parametri **CardCodeBegin** e **CardCodeLength** qui di seguito, e a quello del parametro **RejectShorter** nella sezione *[TimeAttendance]*, vedi a pag. 31) avrebbe successo, se la lunghezza del "RAW data" non corrisponde al valore di questo parametro la carta viene comunque rifiutata. Default: 0 → lunghezza del "RAW data" non controllata.

int **CardCodeBegin**

Posizione iniziale del codice utente nella carta (default 0 → prima posizione).

int **CardCodeLength**

Lunghezza del codice utente a partire dalla posizione **CardCodeBegin** (default 6). La massima lunghezza del codice utente è 16 cifre.

int **EditionBegin**

Posizione iniziale del codice edizione nella carta (default 0 → prima posizione)

int **EditionLength**

Lunghezza del codice utente a partire dalla posizione **EditionBegin**. Valori consentiti:

0 (default) → codice edizione non usato, 1 → codice edizione a singola cifra, 2 → codice edizione a doppia cifra.

int **ShowCardCodeBegin**

Posizione iniziale (all'interno del codice utente già estratto dalla carta) del codice mostrato a display in seguito ad una transazione valida; nel file TRANSACTIONS.TXT viene comunque memorizzato l'intero codice personale, cioè tante cifre quante sono quelle impostate dal parametro **CardCodeLength**. **Nota**: questo parametro ha effetto solo se il successivo parametro **ShowCardCodeLength** è impostato ad un valore diverso da 0 (suo valore di default).

0 (default) → il codice utente viene mostrato a partire dalla prima cifra

n = 1,2.. → le prime *n* cifre del codice utente non vengono visualizzate; la somma dei valori dei parametri **ShowCardCodeBegin** e **ShowCardCodeLength** deve comunque essere inferiore o uguale al valore del parametro **CardCodeLength**, altrimenti la transazione sarà accettata ma nella schermata di conferma non verrà mostrato nessun codice

int **ShowCardCodeLength**

Lunghezza (all'interno del codice utente già estratto dalla carta e a partire dalla posizione **ShowCardCodeBegin**) del codice mostrato a display in seguito ad una transazione valida; nel file TRANSACTIONS.TXT viene comunque memorizzato l'intero codice personale, cioè tante cifre quante sono quelle impostate dal parametro **CardCodeLength**.

0 (default) → il codice utente viene mostrato per intero

n = 1,2.. → vengono mostrate solo *n* cifre del codice utente; la somma dei valori dei parametri **ShowCardCodeBegin** e **ShowCardCodeLength** deve comunque essere inferiore o uguale al valore del parametro **CardCodeLength**, altrimenti la transazione sarà accettata ma nella schermata di conferma non verrà mostrato nessun codice

int **FacilityCodeBegin**

Posizione iniziale del codice comune nella carta.

Default: 0 → prima posizione, 1 → seconda posizione, etc.

int **FacilityCode**

Codice comune. Default : vuoto (codice comune non controllato).

int **Direction**

Determina la direzione di passaggio per le letture effettuate sul lettore relativo alla sezione interessata:

0 → Imposta la direzione fissa "Uscita", indipendentemente dal valore del parametro **DirMode** nella sezione [TimeAttendance]

1 → Imposta la direzione fissa "Entrata", indipendentemente dal valore del parametro **DirMode** nella sezione [TimeAttendance]

2 (default) → La direzione viene determinata in base al valore del parametro **DirMode** nella sezione [TimeAttendance]

3 → Imposta la modalità di selezione della direzione mediante un pulsante collegato ad uno degli ingressi digitali disponibili, definito dal seguente parametro **DirectionInput**: per tutte le letture effettuate sul lettore relativo a questa sezione, se l'ingresso è attivo (cortocircuitato) allora viene automaticamente assegnata la direzione "Entrata", se invece l'ingresso non è attivo (circuito aperto) allora viene automaticamente assegnata la direzione "Uscita", indipendentemente dal valore del parametro **DirMode** nella sezione [TimeAttendance]

int **DirectionInput**

Solo nel caso in cui il precedente parametro **Direction** sia impostato al valore '3', imposta il numero dell'ingresso digitale a cui è collegato il pulsante per la selezione della direzione da assegnare a tutte le letture effettuate sul lettore relativo a questa sezione.

0 (default) → Non utilizzato

1, 2 → input IN1 e IN2 rispettivamente, già disponibili su X1/X2

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1' (vedi §3.4 a pag. 10)

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

int **SkipBioVerify**

In presenza di un modulo biometrico esterno FingerBOX abilitato, disabilita a priori la richiesta di verifica biometrica per tutte le letture di tessera effettuate sul lettore relativo alla sezione interessata. Vedi §11.4 a pag. 135 per ulteriori dettagli.

0 (default) → verifica biometrica richiesta (salvo esenzioni per singoli codici tessera)

1 → verifica biometrica non richiesta

int **DisableFunctions**

Consente di disattivare determinate funzioni del controllo accessi per tutte le letture di tessera effettuate sul lettore relativo alla sezione interessata. Parametro valido a bit.

Bit 0 (+1) → disabilita l'attivazione del relé, indipendentemente dalla direzione associata al lettore

Bit 1 (+2) → disabilita la richiesta del PIN di sicurezza (che ha luogo, se il parametro **AskPin** all'interno della sezione *[AccessControl]* (vedi a pag. 37) è stato impostato a 1, per gli utenti registrati nel file USERS.TXT e aventi un campo PPPP diverso da "0000", vedi §5.9 a pag. 77)

Esempio: impostando il parametro al valore 3 (1+2), vengono disabilitate sia l'attivazione del relé che la richiesta del PIN per tutte le letture effettuate sul lettore relativo alla sezione interessata.

SEZIONE [READER2]

*Nota: tutti i parametri di questa sezione, sono chiamati esattamente e hanno lo stesso significato di quelli delle sezioni **Reader1** e **ExtReader**, ma si riferiscono al lettore secondario sul connettore Molex con dicitura "READER 2" oppure "FINGER BOX" (a seconda della versione hardware). Il parametro **SkipBioVerify** non è presente in quanto questa sezione viene ignorata in presenza di un modulo biometrico esterno FingerBOX abilitato.*

int **CardDecode** (default: 99, ovvero lettore disabilitato, mentre nella sezione *[Reader1]* il default è 30)

int **BaudrateReader**

int **CardLayoutLength**

int **CardCodeBegin**

int **CardCodeLength**

int **EditionBegin**

int **EditionLength**

int **ShowCardCodeBegin**

int **ShowCardCodeLength**

int **FacilityCodeBegin**

int **FacilityCode**

int **Direction**

int **DirectionInput**

int **DisableFunctions**

SEZIONE [EXTREADER]

*Nota: tutti i parametri di questa sezione, eccetto "WiegandOutput" e "ReaderLeds", sono chiamati esattamente e hanno lo stesso significato di quelli delle sezioni **Reader1** e **Reader2** qui sopra, ma si riferiscono al lettore esterno su morsettiera a vite (o agli eventuali lettori aggiuntivi collegati su schede di espansione 914 NeoMAX opzionali, vedi §3.6 a pag. 11, con l'eccezione dei parametri **CardDecode** e **BaudRate**, che in tal caso vengono ignorati poiché la decodifica viene effettuata autonomamente dai 914 NeoMAX ed è fissa all'equivalente del valore '0' di **CardDecode**, cioè lettore di carte magnetiche in traccia 2 o altro tipo di lettore con uscita compatibile).*

int **CardDecode** (default: 99, ovvero lettore disabilitato, mentre nella sezione *[Reader1]* il default è 30)

int **BaudrateReader**

int **CardLayoutLength**

int **CardCodeBegin**

int **CardCodeLength**

int **EditionBegin**

int **EditionLength**

int **ShowCardCodeBegin**

int **ShowCardCodeLength**

int **FacilityCodeBegin**

int **FacilityCode**

int **WiegandOutput**

Presente solo nella sezione [ExtReader]. Consente di abilitare la ritrasmissione automatica di ogni lettura effettuata (su uno qualunque dei 3 lettori diversi collegati) ad un controller remoto nel formato fisso Wiegand 37bit H10302. A tale scopo vengono usati i pin G-LED e R-LED della morsettiera a vite estraibile contrassegnata come "EXTERNAL READER" (oltre al pin GND sullo stesso connettore), i quali pertanto non possono più essere utilizzati per pilotare gli eventuali LED verde e rosso del lettore esterno.

0 → uscita Wiegand 37bit H10302 non abilitata – Gestione dei 2 LED verde e rosso del lettore esterno (default)

1 → uscita Wiegand 37bit H10302 abilitata

int **Direction**

int **DirectionInput**

int **SkipBioVerify**

int **DisableFunctions**

int **ReaderLeds**

Presente solo nella sezione [ExtReader]. Determina il comportamento a riposo del LED rosso del lettore esterno (Nota: funziona solo se **WiegandOutput**=0 in questa stessa sezione).

0 → il LED rosso è normalmente spento, e lampeggia solo in caso di transazione non valida (utile sui lettori RFID3 per mantenere il LED multicolore in stato azzurro a riposo)

1 (default) → il LED rosso è normalmente acceso, e si spegne in caso di transazione valida (mentre il LED verde si accende) oppure lampeggia in caso di transazione non valida

SEZIONE [BIOMETRIC]

int **Enabled**

Attiva la gestione del modulo biometrico esterno FingerBOX per la scansione di impronte digitali

0 (default) → gestione FingerBOX non attivata

1 → attiva la gestione del modulo FingerBOX

int **FreeScan**

Attiva la modalità "autoscan" (o "identificazione 1:N", o "solo dito", vedi §11 a pag. 121 per ulteriori dettagli)

0 (default) → modalità "autoscan" non attivata (funzionamento "solo verifica 1:1")

1 → attiva la modalità "autoscan"

int **SecurityLevel**

Imposta il livello di sicurezza del modulo biometrico. Può assumere valori da 1 a 18 (default 16), con il seguente significato (vedi §11 a pag. 121 per ulteriori dettagli):

1..15: livello fisso → 1: sicurezza minima .. 15:sicurezza massima

16..18: livello variabile automaticamente in base al numero di *template* memorizzati → 16: normale, 17: sicuro, 18: più sicuro)

int **Sensitivity**

Imposta la sensibilità di rilevamento del sensore. Può assumere valori da 1 (sensibilità minima) a 8 (sensibilità massima - default). Con una sensibilità alta il modulo biometrico accetta più facilmente l'impronta immessa, mentre con una minore sensibilità l'immagine dell'impronta immessa sarà più stabile.

int **ImageQuality**

Imposta la qualità dell'immagine scansionata. Può assumere valori da 1 (accetta qualità minima) a 4 (richiede qualità massima), il valore di default è 2. Quando viene scansionata un'impronta, il modulo biometrico controlla se la qualità dell'immagine è adeguata per essere elaborata ulteriormente. Se è scarsa, il modulo biometrico invia un messaggio d'errore. Questo parametro specifica la severità di questo controllo di qualità.

int **LightingCondition**

Imposta le condizioni di luminosità ambientale dei moduli biometrici provvisti di sensore di impronte di tipo ottico (utilizzo all'aperto o al chiuso): in realtà si raccomanda di lasciare sempre questo parametro al valore 0 (default: utilizzo all'aperto), e di non cambiarlo.

int **FastMode**

Imposta la velocità di identificazione 1:N. Può assumere valori da 1 a 7 (default 7), con il seguente significato (vedi §11 a pag. 122 per ulteriori dettagli):

1..6: velocità fissa → 1: normale (più lenta) .. 6: velocità massima

7: velocità variabile automaticamente in base al numero di *template* memorizzati nel modulo

int **MifareFirstBlock**

Imposta il numero (in decimale) del blocco dati a partire dal quale è possibile memorizzare i dati biometrici all'interno di una carta di prossimità Mifare R&W. Questo parametro ha effetto solo se si intende salvare un codice tessera personalizzato e/o i *template* di ciascun utente direttamente all'interno di una carta Mifare personale (vedi §11.2 a pag. 132). I dati vengono in tal caso scritti su tutti i blocchi dati adiacenti disponibili a partire da quello specificato da questo parametro (default 1, cioè il primo blocco disponibile dopo il blocco 0 a sola lettura che contiene anche il codice fisso UID): il codice tessera personalizzato viene scritto a partire dall'offset 1 (quindi escluso il primo byte) per una lunghezza di 8 byte in formato packed BCD (2 cifre per byte), e ad esso seguono i *template* biometrici.

Nota: per usare il codice tessera personalizzato in fase di lettura carta è necessario impostare il parametro **CardDecode** nella sezione relativa al lettore utilizzato al valore '32' (vedi a pag. 42).

int **TemplateSource**

Specifica dove debbano essere cercati i *template* registrati al momento di effettuare l'autenticazione biometrica:

0 → ricerca solo all'interno della carta appena letta. Questa opzione può essere usata solo se si utilizzano carte di prossimità Mifare R&W su ciascuna delle quali sono stati in precedenza memorizzati i *template* del possessore della carta (vedi §11.2 a pag. 132), e solo in modalità "carta + dito": una volta salvata questa impostazione, il parametro **FreeScan** viene automaticamente impostato a 0. Usando un qualunque altro tipo di carta o la digitazione manuale del codice si ottiene sempre il messaggio di errore "**Tessera non valida**".

1 → ricerca solo sul terminale (file USERCODS e memoria interna del modulo FingerBOX). Questa opzione va usata solo se non si vogliono mai utilizzare per la verifica biometrica 1:1 i *template* eventualmente memorizzati su carte di prossimità Mifare R&W.

2 (default) → ricerca prima all'interno della carta appena letta, poi (solo se non trova nulla) sul terminale

3 → ricerca prima sul terminale, poi (solo se non trova nulla) all'interno della carta

int **FreePass**

Disabilita la richiesta di verifica biometrica per ogni lettura (o digitazione, ma solo su X2 e se il parametro **AllowTypeCode**=1 all'interno della sezione *[TimeAttendance]* (vedi §4.11 a pag.31) di un qualunque codice per il quale non sia già stata effettuata una registrazione di impronte: utile nel caso vi sia la necessità di gestire gli accessi di visitatori temporanei. Per ulteriori dettagli si veda il §11.4 a pag. 135.

0 (default) → verifica biometrica sempre richiesta (salvo esenzioni per singoli codici tessera)

1 → verifica biometrica non richiesta per tutti i codici senza registrazione biometrica

int **EnrollAuth**

Se impostato a 1, consente di effettuare la registrazione di impronte ai soli codici tessera già elencati nel file CARDS.TXT in formato "esteso" (cioè con tutti i record di 71 caratteri invece che 69 come nella versione standard) e aventi l'apposito flag **B** (biometrico) a '1'. Per ulteriori dettagli si veda il §5.4 a pag. 71.

Nota: se il successivo parametro **EnrollAll**=1, esso è comunque più prioritario e quindi vengono sempre accettati tutti i codici inseriti.

0 (default) → registrazione di impronte consentita per tutti i codici (in assenza di CARDS.TXT) o per tutti i codici elencati in CARDS.TXT, qualunque sia il formato del file

1 → registrazione di impronte consentita solo ai codici tessera elencati nel file CARDS.TXT in formato "esteso" e aventi flag **B**='1'

int **EnrollAll**

Se impostato a 1, durante le registrazione delle impronte vengono sempre accettati tutti i codici tessera inseriti. Normalmente invece, se è presente almeno un file fra CARDS.TXT e CARDRNGE.TXT, il codice inserito viene accettato solo se è fra quelli elencati in CARDS.TXT o si trova all'interno di un intervallo di codici elencato in CARDRNGE.TXT.

0 (default) → durante la registrazione di impronte viene controllata la presenza del codice inserito nei file CARDS.TXT o CARDRNGE.TXT, se presenti

1 → durante la registrazione di impronte vengono sempre accettati tutti i codici inseriti

int **MinimumQuality**

Imposta il valore minimo del punteggio (*score*) relativo ai *template* affinché essi possano essere memorizzati in fase di registrazione delle impronte. Può assumere valori da 0 a 100, ma si raccomanda di usare valori maggiori o uguali a 70 (default).

Nota: lo *score* non dipende dalla qualità dell'immagine dell'impronta, bensì dal solo contenuto informativo rilevante ai fini del riconoscimento biometrico (*minuzie*) e dalla corrispondenza fra i dati relativi alle due scansioni effettuate in fase di registrazione delle impronte.

int **SaveReaderSource**

Determina se, per ciascuna nuovo utente registrato, il campo **R** presente nei records aggiunti ai file USERCODS.TXT e BIOUPDATE.TXT debba contenere l'identificatore del lettore su cui è stata effettuata la lettura durante l'*enrollment*, oppure debba essere impostato a 0 (che significa sorgente di lettura non considerata).

int **SendTemplate**

Abilita/Disabilita la trasmissione online HTTP immediata di tutti i nuovi record memorizzati nel file BIOUPDATE.TXT, che possono essere relativi a qualunque operazione effettuata all'interno del menu di gestione dell'archivio di impronte: registrazione di ogni nuova impronta (inclusi i dati dei *template*), cancellazione utente, cambio degli attributi di un utente (esenzione dalla verifica biometrica, impostazione dei diritti di amministratore).

0 → Trasmissione online disabilitata

1 (default) → Trasmissione online abilitata

SEZIONE [SYSTEM]

int **TurnOffTimeout**

Timeout di inattività nel funzionamento a batteria (in minuti) - Default 10 minuti

int **TurnoffBacklight**

Abilita/disabilita lo spegnimento immediato della retroilluminazione del display durante il funzionamento a batteria.

1 (default) → Spegnimento della retroilluminazione abilitato

0 → Spegnimento disabilitato (lo schermo rimane retroilluminato anche in assenza di alimentazione)

int **Backlight**

Abilita/disabilita la retroilluminazione del display durante il funzionamento in presenza di alimentazione.

1 (default) → Retroilluminazione abilitata

0 → Retroilluminazione disabilitata anche in presenza di alimentazione

int **Contrast**

Contrasto del display (0..9) - Default 5

int **TurnoffEthernet**

Abilita/disabilita la disattivazione immediata della scheda di rete durante il funzionamento a batteria, allo scopo di prolungarne la durata.

1 → Disattivazione della scheda di rete abilitata

0 (default) → Disattivazione della scheda di rete disabilitata

int **VirtualKeyIn1**

Consente di utilizzare l'attivazione dell'ingresso digitale IN1 per emulare la pressione di un tasto funzione:

0 (default) → non usato

1 → tasto [↔] (inversione direzione)

3 → tasto [Clr]

4 → tasto ▲

5 → tasto ▼

6 → tasto ↵ (Invio)

int **VirtualKeyIn2**

Come **VirtualKeyIn1**, ma relativamente all'ingresso digitale IN2.

int **LogLevel**

Determina quanti eventi, a seconda della loro importanza, vengono registrati nel file LOG.TXT, e la corrispondente dimensione massima di un singolo file di log (ogni volta che il file LOG.TXT eccede la dimensione massima, viene automaticamente rinominato in LOG.0.TXT, e gli eventuali file precedenti (se esistono) vengono a loro volta rinominati, fino al file più vecchio mantenuto in memoria che è sempre LOG.3.TXT).

0: molto dettagliato (tutti gli eventi vengono registrati) - dimensione massima: 10MB

1: dettagliato - dimensione massima: 5MB

2 (default): solo gli eventi principali vengono registrati - dimensione massima: 2MB

3: file LOG.TXT quasi inutilizzato (vengono registrati solo i riavvii e gli errori della micro-SD - dimensione massima: 1MB)

Nota: se questo parametro viene impostato ad un valore minore di 2, il terminale continua a registrare in maniera dettagliata per un tempo massimo di una settimana, dopodiché torna a registrare solo gli eventi principali, come se il parametro fosse tornato al valore 2. E' sufficiente riavviare il terminale o effettuare un salvataggio dei parametri prima che termini la settimana per prolungare automaticamente il periodo di registrazione dettagliata di un'altra settimana. Allo scadere di tale periodo, nella pagina "**System**" del web server HTTP del terminale e alla voce "**Log Level**" viene effettivamente mostrato il valore 2, anche se in realtà il valore del parametro all'interno del file PARAMETERS.TXT non viene cambiato. In questa fase, effettuando un salvataggio dei parametri il valore 2 viene definitivamente scritto nel file; se invece si riavvia semplicemente il terminale è possibile iniziare un nuovo periodo di registrazione dettagliata: da quel momento, e per un'altra settimana, nella pagina "**System**" del web server HTTP del terminale e alla voce "**Log Level**" viene nuovamente mostrato il valore ancora impostato nel file (0 o 1).
Questo meccanismo consente di evitare registrazioni continue sulla micro-SD per tempi molto lunghi, che potrebbero facilmente causarne la corruzione precoce (il numero di operazioni di scrittura su questo tipo di memorie non è illimitato, anche se molto grande). In ogni caso, per sicurezza, si consiglia di impostare valori di **LogLevel** inferiori a 2 solo per brevi periodi ed esclusivamente a scopo di debug, e ricordarsi di reimpostarlo al valore 2 una volta terminato il periodo di osservazione.

Int **LogAggressiveFlush**

Attiva la registrazione immediata di ogni messggio di log (senza effettuare bufferizzazione).

0 (default): i messaggi di log vengono prima bufferizzati e quindi registrati a gruppi

1: ciascun messaggio di log viene registrato non appena viene generato – da usare solo per brevi periodi e a scopo di debug

int **TTY1Config**

Flag riservato usato dal programma Xatl@s: quando si connette la prima volta al terminale, Xatl@s controlla questo flag, e se trova il valore 1 (default) invia la configurazione completa dei parametri come definita nel programma, e alla fine imposta il flag a 0.

Attenzione: assicuratevi che questo parametro sia sempre impostato al valore di default (1) <u>in tutti i casi in cui non viene effettivamente utilizzato il programma XAtlas</u>, poiché in caso contrario potrebbero verificarsi comportamenti anomali.

string **Language**

Lingua del terminale. Le lingue sono memorizzate nel file LANGUAGE.TXT. Questo parametro seleziona uno degli identificatori di lingua all'interno del file. Potete aggiungere le lingue che preferite, e anche cambiare I messaggi di default usando questo file.

Se il parametro non è specificato vengono usati i messaggi di default (in inglese) presi direttamente dal firmware del terminale.

int **FontEncoding**

Non più utilizzato (vedi §20 a pag. 177)

string **SecureOperatorPassword**

Contiene il valore di *hash* relativo alla password numerica correntemente impostata per l'accesso al menu supervisore dalla tastiera del terminale (default "00000"), calcolato mediante un algoritmo non invertibile (**PBKDF2**) a cui si applica una chiave ("*salt data*") fissa e riservata: si tratta di una stringa di 64 cifre esadecimali (32 bytes). Per ulteriori informazioni fare riferimento al §17.3 a pag. 162.

Default: 6D8B9EF12C9B40ECBEC4109BB0479F9B17353E8A64BFB50DA0074803A6FCD304

string **RemoteUsernameCrypto**

Contiene il valore di *hash* relativo al nome utente dell'account amministratore correntemente impostato per l'accesso remoto da un client FTP o browser web (default "admin"), calcolato mediante un algoritmo non invertibile (**PBKDF2**) a cui si applica una chiave ("*salt data*") fissa e riservata: si tratta di una stringa di 64 cifre esadecimali (32 bytes). Per ulteriori informazioni fare riferimento al §17.3 a pag. 162.

Default: 4386CF3B44F59C866B7F8006CA5A75B571721A965706259D57F7500171051E19

string **SecureRemotePassword**

Contiene il valore di *hash* relativo alla password dell'account amministratore correntemente impostata per l'accesso remoto da un client FTP o browser web (default "admin"), calcolato mediante un algoritmo non invertibile (**PBKDF2**) a cui si applica una chiave ("*salt data*") fissa e riservata: si tratta di una stringa di 64 cifre esadecimali (32 bytes). Per ulteriori informazioni fare riferimento al §17.3 a pag. 162.

Default: 4386CF3B44F59C866B7F8006CA5A75B571721A965706259D57F7500171051E19

string **ManagerUsernameCrypto**

Contiene il valore di *hash* relativo al nome utente dell'account CLOKI correntemente impostato per l'accesso remoto solo da browser web (default "manager"), calcolato mediante un algoritmo non invertibile (**PBKDF2**) a cui si applica una chiave ("*salt data*") fissa e riservata: si tratta di una stringa di 64 cifre esadecimali (32 bytes). Per ulteriori informazioni fare riferimento al §17.3 a pag. 162.

Default: CD245F818F8E477B9BD47CC221142F043B65100070BC9F05A6F8C8CF06CBA151

string **SecureManagerPassword**

Contiene il valore di *hash* relativo alla password dell'account CLOKI correntemente impostata per l'accesso remoto solo da browser web (default "manager"), calcolato mediante un algoritmo non invertibile (**PBKDF2**) a cui si applica una chiave ("*salt data*") fissa e riservata: si tratta di una stringa di 64 cifre esadecimali (32 bytes). Per ulteriori informazioni fare riferimento al §17.3 a pag. 162.

Default: CD245F818F8E477B9BD47CC221142F043B65100070BC9F05A6F8C8CF06CBA151

string **TimeLock**

Data di scadenza del periodo di valutazione del terminale, nel formato AAAAMMGG. Se la data corrente risulta essere maggiore o uguale a questo valore, il terminale mostra un messaggio di avvertimento (definito dalla stringa personalizzabile **STR_19** nel file LANGUAGE.TXT, vedi §8 a pag. 105) e richiede l'inserimento di un codice di sblocco costituito dalla sequenza delle 4 cifre della data corrente (nell'ordine GGMM) in complemento a 9. Una volta sbloccato con questa procedura, il terminale riparte reimpostando automaticamente il parametro **TimeLock** al valore di default (vuoto). E' anche possibile sbloccare il terminale da remoto reimpostando il parametro **TimeLock** ad un valore superiore (per estendere il periodo di prova) o al valore di default (vuoto).

int **AudioVolume**

Imposta il volume del segnalatore acustico integrato. Nota: il volume si può impostare solo editando il file PARAMETERS.TXT oppure, analogamente, mediante il menu a tendina "**Audio volume**" nella pagina "**System**" del web server HTTP del terminale. Non è possibile modificarlo dalla console del terminale poiché manca la relativa opzione nel menu supervisore.

1 (default): volume alto

2: volume medio

3: volume basso

int **TTY1Legacy**

Flag riservato usato dal programma Xatl@s. Default: 1

int **TTY1Busy**

Flag riservato usato dal programma Xatl@s. Default: 0

int **CryptoEnabled**

Consente di abilitare l'utilizzo della crittografia per i file che contengono dati personali degli utenti: i file USERS.TXT, BIOUPDATE.TXT, BIODATA.TXT non verranno più utilizzati (come se non ci fossero), e al loro posto dovranno essere utilizzati i file USERS.EF*x*, BIOUPDATE.EF*x*, BIODATA.EF*x* (dove il carattere *x* dipende dal tipo di cifratura); ogni nuovo record caricato dal server in questi file dovrà essere in formato già criptato in base alla chiave di cifratura scelta(*) e ad un algoritmo invertibile (**RC4 + Base64**), e ogni nuovo record generato autonomamente dal terminale durante una registrazione biometrica o in fase di esportazione dell'archivio biometrico verrà scritto in formato criptato, sempre in base a tale chiave(*).

0 (default): crittografia disabilitata

1: crittografia abilitata

(*)Affinché sia possibile abilitare la crittografia è necessario avere già impostato la necessaria chiave di cifratura, scelta a piacere e memorizzata solamente nella memoria RAM tamponata dell'orologio del terminale (quindi non nel file system del terminale). Al default, la chiave di cifratura è impostata ad un valore non valido, e in questo stato non vengono applicati cambiamenti a questo parametro. Per ulteriori informazioni fare riferimento al §17.2 a pag. 159.

int **TrnsHash**

Consente di abilitare l'aggiunta di un campo "valore di *hash*" a ciascun record nei file TRANSACTIONS.TXT (vedi §7 a pag. 96) e btransaction.loc. Se il file btransaction.loc corrente è stato generato con una versione di fw precedente alla **g01_build***nnn* e vi sono transazioni ancora "pendenti" (cioè non ancora ricevute e/o non confermate da un server HTTP), entrambi i file continuano ad essere registrati nel formato standard: solo quando il server avrà ricevuto (e confermato) tutte le transazioni già presenti nel file btransaction.loc, oppure in seguito ad una cancellazione manuale o alla rinomina automatica di tale file al verificarsi di una delle condizioni descritte in dettaglio al §7 a pag. 96, il file btransactions.loc verrà ricreato e le transazioni / eventi inizieranno ad essere registrati in entrambi i file con l'aggiunta del valore di *hash*, calcolato a partire dai dati essenziali di ciascuna transazione / evento (codice personale, data, ora, direzione di transito e indirizzo MAC del terminale su cui è stata effettuata) mediante un algoritmo non invertibile (**PBKDF2**) e usando come chiave ("*salt data*") la chiave di cifratura scelta(*), che è la stessa utilizzata anche per la crittografia dei dati personali degli utenti (vedi par. **CryptoEnabled** qui sopra). Questo non viene fatto con lo scopo di nascondere i dati delle transazioni (i quali, contenendo solo codici, non stabiliscono un'associazione univoca con gli utenti), bensì con lo scopo di consentire a posteriori un controllo di congruenza delle transazioni stesse: in questo modo è possibile identificare eventuali timbrature aggiunte fraudolentemente ai file.

0 (default): *hash* di transazioni ed eventi disabilitato

1: *hash* di transazioni ed eventi abilitato

(*)Affinché sia possibile abilitare l'aggiunta del valore di *hash* alle transazioni / eventi è necessario avere già impostato la necessaria chiave di cifratura, scelta a piacere e memorizzata solamente nella memoria RAM tamponata dell'orologio del terminale (quindi non nel file system del terminale). Al default, la chiave di

cifratura è impostata ad un valore non valido, e in questo stato non vengono applicati cambiamenti a questo parametro. Per ulteriori informazioni fare riferimento al §17.4 a pag. 163.

string **FirmwareKey**

Chiave di attivazione per funzioni opzionali del firmware. Vedi §4.12 a pag. 65 per una descrizione dettagliata. Default: vuoto.

string **Firmware**

Stringa identificativa della versione corrente del firmware nel formato "a*NN*build*nnnn*".

SEZIONE [TIMESETTINGS]

Int	**SMonth**	1..12	Default: 99 (cioè "non usato")
Int	**SDay**	1..31	Default: 99 (cioè "non usato")
Int	**SHour**	0..23	Default: 99 (cioè "non usato")
Int	**SMin**	0..59	Default: 99 (cioè "non usato")

*I parametri qui sopra definiscono la data e ora di passaggio all'ora legale. Questa impostazione è necessaria se tali valori non coincidono con quelli utilizzati per default nel caso di cambio automatico dell'ora, o se il cambio automatico è disabilitato (vedi parametro **AutoDayLightSavingTime** qui sotto).*

Int	**WMonth**	1..12	Default: 99 (cioè "non usato")
Int	**WDay**	1..31	Default: 99 (cioè "non usato")
Int	**WHour**	0..23	Default: 99 (cioè "non usato")
Int	**WMin**	0..59	Default: 99 (cioè "non usato")

I parametri qui sopra definiscono la data e ora di passaggio all'ora solare. Valgono le stesse considerazioni fatte per i parametri che definiscono il passaggio all'ora legale.

int **AutoDayLightSavingTime**

0 → Impostazione manuale delle date di passaggio all'ora legale/solare (definite dai parametri elencati sotto)

1 (default) → Imposta automaticamente i passaggi all'ora legale/solare secondo le regole europee: alle 01:00 dell'orario UTC/GMT (quindi alle 02:00 secondo l'orario vigente in Italia CET=UTC/GMT+1) dell'ultima domenica di marzo l'orologio viene portato avanti di un ora per il passaggio all'ora legale (DST), mentre alle 02:00 dell'orario UTC/GMT (quindi alle 03:00 secondo l'orario vigente in Italia CET=UTC/GMT+1) dell'ultima domenica di ottobre l'orologio viene portato indietro di un ora per il ritorno all'ora solare. ***Nota***: affinché il cambio orario automatico avvenga all'ora corretta, dovete ricordarvi di impostare il parametro **UTC** qui sotto al valore relativo al vostro fuso orario (ad esempio per l'Italia dovete impostare il valore "+01:00").

2 → Imposta automaticamente i passaggi all'ora legale/solare secondo le regole nordamericane: alle 02:00 dell'orario locale (qualunque sia il fuso orario) della seconda domenica di marzo l'orologio viene portato avanti di un ora per il passaggio all'ora legale (DST), mentre alle 02:00 dell'orario locale (qualunque sia il fuso orario) della prima domenica di novembre l'orologio viene portato indietro di un ora per il ritorno all'ora solare.

***Nota**: in entrambi i casi citati qui sopra, anche se X1/X2 è spento nel momento in cui il cambio dovrebbe avere luogo, il cambio di orario avviene correttamente non appena acceso il terminale. Se non siete sicuri di avere effettuato le impostazioni corrette, dopo averle salvate potete controllare, nella pagina "**Daylight Saving Time**" del web server http del terminale, la data e ora in cui risulta essere schedulato il prossimo cambio orario ("**Next shift**"): si consiglia di usare Ctrl+F5 sulla vostra tastiera per essere sicuri di visualizzare i dati aggiornati.*

int **RecordDayLightSaving**

1 → Aggiunge il campo DAYLIGHT nel file TRANSACTIONS.TXT file, che specifica se la transazione è stata effettuata durante l'ora solare o legale:

0= ora solare, 1= ora legale.

0 (default) → campo DAYLIGHT vuoto

string **UTC**

Differenza fra il fuso orario locale e quello universale UTC/GMT, il formato è +HH:MM o –HH:MM. Default: +00:00

***Attenzione**: il parametro **UTC** deve essere necessariamente impostato al valore relativo al vostro fuso orario (ad esempio per l'Italia dovete impostare il valore "+01:00") nei seguenti casi:*

*1) se avete impostato il passaggio automatico all'ora legale/solare secondo le regole europee (vedi parametro **AutoDayLightSavingTime** qui sopra), altrimenti il cambio orario non avverrebbe all'ora corretta;*

*2) se avete impostato l'aggiornamento automatico dell'orario tramite server di sincronizzazione NTP (vedi parametro **UseNTP** qui sotto), altrimenti il terminale verrebbe impostato all'orario universale UTC/GMT invece che che a quello locale del vostro fuso orario.*

int **RecordUTC**

1 → Aggiunge il campo UTC nel file TRANSACTIONS.TXT usando il fuso orario specificato dal parametro **UTC** (vedi sopra)

0 (default) → il campo UTC è vuoto

int **UseNTP**

0 (default) → non vengono effettuati aggiornamenti automatici della data/ora da server di sincronizz. NTP

1 → Viene periodicamente effettuato un aggiornamento automatico della data/ora, inviando una richiesta SNTP al server di sincronizzazione NTP specificato dal successivo parametro **NTPServerName** (vedi 4.1 a pag. 16). ***Attenzione***: affinché l'orario venga impostato correttamente, dovete ricordarvi di impostare il parametro UTC qui sopra al valore relativo al vostro fuso orario (ad esempio per l'Italia dovete impostare il valore "+01:00").

string **NTPServerName**

Indirizzo IP o URL logico del server di sincronizzazione orario NTP desiderato. Default: "pool.ntp.org"

int **NTPRefresh**

Intervallo (espresso in secondi) fra una richiesta di sincronizzazione SNTP e quella successiva, in assenza di riavvii del terminale o di aggiornamenti dei parametri di configurazione (infatti data e ora vengono comunque sincronizzati circa 30 secondi dopo il verificarsi di tali eventi). Default: 604800 (che equivale a una settimana).

SEZIONE [ETHERNET]

int **DHCP**

1 (default): DHCP ON

0: DHCP OFF

string **IPAddress**

Valore di default 192.168.1.240 (che sarebbe usato solo se il DHCP fosse OFF)

string **Gateway**

Gateway, default 0.0.0.0

string **Subnet**

Subnet mask, default 255.255.255.0

string **Primary_DNS**

Indirizzo DNS primario, default 0.0.0.0

string **Secondary_DNS**

Indirizzo DNS secondario, default 0.0.0.0

int **FtpPort**

Porta usata per le comunicazioni dal server FTP del terminale. ***Nota***: non ha effetto sul comportamento del client FTP del terminale per l'invio automatico delle transazioni (a tale scopo si può usare il parametro **ServerUrl** nella sezione *[FtpClient]* qui sotto). Default: 21.

int **HttpPort**

Porta usata per le comunicazioni dal server HTTP del terminale. ***Nota***: non ha effetto sul comportamento del client HTTP del terminale per la comunicazione in online (a tale scopo si può usare il parametro **MasterUrl** qui sotto).. Default: 80.

string **MasterUrl**

Indirizzo IP o URL logico dell'host HTTP. Se l'host HTTP è in ascolto su una porta diversa dal default (porta 80), è possibile specificare la porta da utilizzare subito dopo l'indirizzo stesso dell'host con il server HTTP. Default vuoto. Formato:

<indirizzoIP_o_URL>:*<porta>* (Esempio 192.168.1.1:8080)

int **Protocol**

Imposta il protocollo per la comunicazione col server:

0 (default): da usare nel caso in cui X1/X2 venga gestito dal programma Xatl@s, oppure nel caso si intenda usare solo il protocollo FTP per configurare o scaricare in modalità *batch* le transazioni registrate in offline dal terminale.

1: attiva la gestione del protocollo HTTP, vedi §12 a pag. 137. Il protocollo FTP è comunque sempre attivo.

string **httpOnlineMessage**

Vedi §12 a pag. 137 per una descrizione dettagliata

string **httpBatchMessage**

Vedi §12 a pag. 137 per una descrizione dettagliata

string **httpKeepAliveMessage**

Vedi §12 a pag. 137 per una descrizione dettagliata

string **TermID**

default "X1" (o "X2", a seconda del modello del terminale) – Identificatore unico del terminale. L'uso dell'indirizzo MAC è sconsigliato perchè l'indirizzo MAC è l'unica cosa che cambia in caso di sostituzione del terminale (vedi "Introduzione" a pag. 5)

int **ConnTimeout**

Timeout in collegamento ONLINE, default 5 secondi

int **KeepAliveInterval**

Vedi §12 a pag. 137 per una descrizione dettagliata

int **RetryTimeout**

Timeout (espresso in secondi) di riconnessione al server TTY1 (riservato al programma Xatl@s). Default: 60.

int **EnableHTTPServer**

1 (default): Server HTTP abilitato

0: Server HTTP disabilitato

int **EnableFTPServer**

1 (default): Server FTP abilitato

0: Server FTP disabilitato

Nota: dopo avere disabilitato un server, le nuove connessioni verranno rifiutate, mentre quelle già attive possono ancora essere utilizzate.

int **EncodeUrl**

Questo parametro consente di abilitare la codifica *percent-encoding* (che consiste in un carattere '%' seguito da 2 cifre esadecimali) per alcuni caratteri (quelli normalmente non consentiti negli URL) che possono trovarsi all'interno dei messaggi HTTP GET trasmessi dal terminale quando è attiva la gestione del protocollo HTTP (vedi 12 a pag. 137). A seconda del tipo di server HTTP utilizzato, questa impostazione può essere necessaria per garantire una corretta interoperabilità fra i dispositivi.

I caratteri consentiti, che non vengono mai sottoposti a *percent-encoding* sono i seguenti: 'A'..'Z', 'a'..'z', '0'..'9', '-', '_', '.', '~'

Alcuni caratteri non alfanumerici vengono (opzionalmente) sottoposti a *percent-encoding* solo se si trovano all'interno dei campi variabili contenuti nei messaggi HTTP GET (ad esempio nel valore di un parametro, o all'interno di un record relativo ad una transazione), mentre rimangono sempre invariati quando vengono usati nel loro ruolo di caratteri riservati per identificare le varie parti dell'URL ed i *server tag* in esso contenuti, ad esempio '/', '?', '=', '&', ',' ecc. Ad esempio, le virgole ',' normalmente presenti all'interno dei record in formato standard relativi alle transazioni vengono automaticamente convertite nella sequenza "%2C" se questo parametro viene impostato ad 1.

Tutti gli altri caratteri non alfanumerici vengono sempre sottoposti a *percent-encoding* se questo parametro viene impostato ad 1: spazio ' ' ("%20"), *pipe* '|' ("%7C") ecc.

0 (default): *percent-encoding* disabilitato per tutti i caratteri, ad eccezione dei caratteri non alfanumerici (come ad esempio gli spazi ' ') eventualmente contenuti nel campo corrispondente al *server tag* **$termid$** e nelle risposte ai comandi HTTP **GETPAR** e **RDR** (vedi §12.1 a pag. 137 e §12.4 a pag. 141)

1: *percent-encoding* abilitato

Nota: nel caso in cui venga impostato il par. **TTY1Legacy**=0 nella sezione *[System]* (modalità "Xatl@s", vedi pag. 53), il *percent-encoding* viene comunque abilitato, a prescindere dal valore del parametro **EncodeUrl**.

SEZIONE [GPRS]

int **Enabled**

Attiva la gestione del modem GPRS integrato opzionale.

0 (default): Modem GPRS disabilitato

1: Modem GPRS abilitato

int **ConnectionInterval**

Imposta l'intervallo di tempo (in minuti) fra una connessione e quella successiva.

0 (default): il modem rimane sempre collegato una volta effettuata la connessione GPRS al fornitore di servizi (valore consigliato in caso di connessione al server FTP del terminale)

9999: il modem effettuerà la connessione GPRS solo se vi sono delle schedulazioni relative a connessioni GPRS e/o esportazioni da client FTP, da impostare mediante il file ALARMS.TXT (vedi §4.2 a pag. 18). In questo caso la connessione GPRS avviene solo in corrispondenza degli orari impostati, e viene chiusa automaticamente al termine di ciascuna esportazione da FTP client, o agli orari delle eventuali disconnessioni schedulate (se specificate).

Se impostato ad un valore diverso da 0 o 9999, invece, ogni connessione dura solo 5 minuti, trascorsi i quali viene valutato se vi sia in quel momento un'attività di comunicazione online significativa: in caso contrario la connessione GPRS viene interrotta, e tale rimane per un tempo pari al valore di questo parametro. Allo scadere dell'intervallo viene effettuata una nuova connessione che dura solo 5 minuti, e così via (per ulteriori dettagli si veda il §15 a pag. 152).

In ogni caso, un host HTTP può forzare la chiusura connessione in qualunque momento usando il tag "**gprs=off**" nella risposta ai pacchetti "KeepAlive" ricevuti, vedi §12.4 a pag. 141.

string **ATextraCommand**

Comando speciale per il modem GPRS che contiene il nome del punto di accesso alla rete (APN, *Access Point Name*): si tratta di un parametro fondamentale per il funzionamento della connessione GPRS.

Default: vuoto. Normalmente potete impostarlo al valore seguente:

AT+CGDCONT=1,IP,*<APN>***,,0,0**

dove *<APN>* è una stringa contenente il nome del punto di accesso, che dipende dal fornitore di servizi scelto. Ad esempio, per l'Italia, per la rete Vodafone *<APN>*=**web.omnitel.it** mentre per la rete TIM *<APN>*=**ibox.tim.it**

Nota: questo campo non deve contenere delle virgolette ""

string **Dialnum**

Numero telefonico da chiamare per collegarsi alla rete GPRS: si tratta di un parametro fondamentale per il funzionamento della connessione GPRS.

Default: vuoto. Normalmente potete impostarlo al valore ***99***1#**

Se così non dovesse funzionare potete provare col valore ***99#**

string **User**

Nome utente per effettuare l'accesso alla rete GPRS, solo se richiesto dal fornitore di servizi scelto.

Default: vuoto

string **Password**

Password per effettuare l'accesso alla rete GPRS, solo se richiesto dal fornitore di servizi scelto.

Default: vuoto

SEZIONE [FTPCLIENT]

string **ServerUrl**

Contiene l'indirizzo del server FTP a cui inviare il file TRANSACTIONS.TXT corrente (vedi §7.3 a pag. 102) agli orari schedulati tramite il file ALARMS.TXT (vedi §4.2 a pag. 18). Può essere sia un indirizzo IP che un URL logico. In caso il server FTP sull'host stia comunicando su una porta differente dal default (porta 21), è possibile specificare la porta da utilizzare subito dopo l'indirizzo stesso dell'host con il server FTP. Default vuoto. Formato:

<indirizzoIP_o_URL>:*<porta>* (Esempio ftp.axesstmc.com:21)

string **User**

Contiene il nome utente da utilizzare per l'autenticazione al server FTP

Default vuoto

string **Password**

Contiene la password da utilizzare per l'autenticazione al server FTP

Default vuoto

int **PassiveMode**

Attiva la connessione in modalità passiva. Può essere richiesta per alcuni server FTP

0: Modalità passiva disabilitata

1 (default): Modalità passiva abilitata

int **RetryNumber**

Indica il numero di tentativi di connessione FTP da ripetere in caso di errore durante le operazioni di upload. Tra un tentativo e l'altro viene atteso un intervallo di tempo casuale, per evitare la concomitanza dei tentativi tra più terminali aventi lo stesso tipo di schedulazione.

Default : 3

int **ResponseTimeout**

Se la connessione al server FTP non è stata stabilita con successo dopo questo intervallo di tempo (in secondi, default: 10), il tentativo di connessione viene interrotto.

string **DestinationFileName**

Indica il nome del file in cui verranno salvate le transazioni inviate all'interno del server FTP. E' possibile salvare il file in una sottocartella (che deve essere già stata creata) sul server specificandone il nome completo. Se il campo contiene solo il nome del file, questo sarà memorizzato nella cartella principale del server FTP.

Default : "TRANSACTIONS.TXT". Lunghezza max: 63 caratteri

int **FileOpeningMode**

Indica la modalità di apertura del file.

0 : Modalità accodamento: se il file esiste già sul server, le nuove transazioni vengono accodate a quelle eventualmente già presenti (default)

1 : Modalità nuovo file: ad ogni invio schedulato verrà creato un nuovo file, il cui nome ha la seguente sintassi: "AAAAMMGG-hhmmss_***DestinationFileName***", dove AAAAMMGG e hhmmss sono rispettivamente la data e l'ora dell'invio. Il nuovo file conterrà solo le transazioni del file TRANSACTIONS.TXT corrente; se non ci sono nuove transazioni al momento dell'invio (il che significa che non è ancora stato creato un nuovo file TRANSACTIONS.TXT sul terminale), non verrà creato nessun file neppure sul server.

SEZIONE [USB]

int **Enabled**

Abilita la gestione delle chiavette di memoria USB formattate in FAT32 (solo su versioni di hardware **006** e successive, vedi §14 a pag. 148).

0 (default): Gestione USB disabilitata

1: Gestione USB abilitata

string **SecurePasswordUSB**

Contiene il valore di *hash* relativo alla password numerica correntemente impostata per l'accesso al menu di gestione delle chiavette di memoria USB in seguito all'inserimento della chiavetta (default "00000"), calcolato mediante un algoritmo non invertibile (**PBKDF2**) a cui si applica una chiave ("*salt data*") fissa e riservata: si tratta di una stringa di 64 cifre esadecimali (32 bytes). Per ulteriori informazioni fare riferimento al §17.3 a pag. 162

Default: 6D8B9EF12C9B40ECBEC4109BB0479F9B17353E8A64BFB50DA0074803A6FCD304

string **TrnsFileUSB**

Nome del file contenente le transazioni che verrà creato sulla chiavetta USB (il default è "TRANSACTIONS.TXT" come sul terminale). **Nota**: non è possibile definire un percorso per salvare il file all'interno di una qualunque cartella diversa dalla *root* della chiavetta USB.

int **MoveTrnsToUSB**

Specifica se le transazioni debbano essere "spostate" a tutti gli effetti, il che significa che il file TRANSACTIONS.TXT corrente verrà anche rimosso dal terminale, rinominandolo "TRANSACTIONS.0.TXT" e creandone uno nuovo secondo il meccanismo descritto al §4.10 a pag. 32 relativamente al funzionamento del parametro **DeleteOld**=1, oppure semplicemente "copiate".

0: Non sposta le transazioni, limitandosi a copiarle e lasciando inalterato il file TRANSACTIONS.TXT

1: Sposta le transazioni rinominando il precedente file TRANSACTIONS.TXT (default)

SEZIONE [PRINTER]

int **Enabled**

Abilita la gestione della stampante (solo su versioni di hardware **011** e successive, vedi §3.8 a pag. 13).

0 (default): Gestione stampante disabilitata

1: Gestione stampante abilitata

int **Baudrate**

Default 9600. Altri valori ammessi: 19200, 38400, 57600.

Note:

1) questa sezione è presente solo sulla versione speciale di firmware Aperio (vedi §18 a pag. 165);

2) tutti i parametri di questa sezione non devono essere modificati manualmente: essi saranno automaticamente aggiornati in seguito all'inserimento della stringa di configurazione Mifare criptata relativa al vostro specifico impianto di serrature wireless offline Aperio, da effettuarsi mediante la casella "Command" disponibile nella pagina "Reader 1" del web server http, oppure caricando un file chiamato READER1.TXT (con una sola linea contentente la stringa di configurazione Mifare criptata), e quindi cambiando un qualunque parametro (o semplicemente riavviando il terminale);

3) l'opportuna configurazione di tutti i parametri di questa sezione è necessaria per consentire ogni tipo di operazione sulle tessere fornite con le serrature wireless Aperio offline;

4) la semplice copia di questa sezione su un altro terminale è inutile, in quanto i parametri sono criptati in modo diverso su diversi terminali: le impostazioni non congruenti sarebbero rilevate e tutti i parametri di questa sezione sarebbero reimpostati al loro valore di default, che a tutti gli effetti significano "terminale non configurato".

string **KeyA**

Contiene una rappresentazione criptata della chiave "A" di autenticazione da usare per tutte le tessere Mifare in un impianto di serrature wireless Aperio offline. Il valore di questo parametro è sempre diverso su diversi terminali, anche se l'effettiva chiave "A" utilizzata è la stessa.

string **KeyB**

Contiene una rappresentazione criptata della chiave "B" di autenticazione da usare per tutte le tessere Mifare in un impianto di serrature wireless Aperio offline. Il valore di questo parametro è sempre diverso su diversi terminali, anche se l'effettiva chiave "B" utilizzata è la stessa.

string **SiteCode**

Contiene una rappresentazione criptata del codice impianto univoco scritto all'interno di qualunque tessera da usare in un impianto di serrature wireless Aperio offline. Il valore di questo parametro è sempre diverso su diversi terminali, anche se l'effettivo codice impianto utilizzato è lo stesso.

string **Integrity**

Consente di testare la congruenza dei precedenti parametri criptati **KeyA**, **KeyB** e **SiteCode**. Il valore di questo parametro è sempre diverso su diversi terminali.

int **StandardSector**

Contiene il numero del settore, all'interno di qualunque tessera da usare in un impianto di serrature wireless Aperio offline, che è stato programmato come "standard sector", ovvero il settore protetto che contiene le informazioni relative al codice impianto e la data e ora di inizio e fine validità.

string **AccessMatrix**

Contiene l'indicazione di quali settori, all'interno delle ACCESS CARDS da usare in un impianto di serrature wireless Aperio offline, sono stati programmati come "matrice degli accessi", ovvero i settori protetti che contengono le informazioni relative a quali serrature sono abilitate all'accesso per ciascuna tessera.

string **AlarmsSector**

Contiene l'indicazione di quali settori, all'interno delle ACCESS CARDS da usare in un impianto di serrature wireless Aperio offline, sono stati programmati come "settori allarmi", ovvero i settori protetti che contengono le informazioni relative agli eventi (tipicamente, sia i tentativi di accesso validi che quelli non validi).

Ecco qui di seguito il contenuto del file PARAMETERS.TXT file (con tutti i parametri al valore di default) che viene creato automaticamente formattando la micro-SD card oppure cancellando il file PARAMETERS.TXT attualmente in uso:

```
[TimeAttendance]
SecondsShown=1
```

AmPm=0
MonthDay=0
DateSeparator=47
DirMode=4
BeepOk=100
BeepError=99
CompanyName=
ShowCode=2
RejectShorter=1
AllowTypeCode=0
DisableReviewTA=0
ReviewDaysTA=30
DisableTypeCodeReviewTA=0
Offline=3
MaxPendingRecord=12000
RepeatTimeOut=0
DeleteOld=0
CustomRecord=””
CustomEntry=”1”
CustomExit=”0”
BeepOnCard=0
ScreenOk=""
ScreenError=""
HideTypedCode=0
MultiFormat=0
MandatoryFunction=0
EnableFastMenu=1
Payload=0
AllowTypeReason=0

[AccessControl]
Enabled=0
RelayActivation=5
EntryRelay=1
ExitRelay=1
DeniedRelay=0
DeniedRelayTimeout=5
Indexing=0
PinOnly=0
AskPin=0
FullTable=0
RecordInvalidAccess=0
EnableNeoMaxI/O=1
GateEnabled=0
GateType=0
TimeOutOpen=50
TimeOutOpenExtended=100
TimeOutClose=50
TimeOutCloseExtended=100
ManualUnlockIN=0
ManualUnlockOUT=0
Emergency=0
EmergencyRelay=0
GateLocked=0

GateLockedRelay=0
GateAlert=0
GateTransitOk=0
InterLocked=0
GateBusy=0
GateSensor1=1
GateSensor2=0
GateState1=1
GateState2=1
ExternalNoTransit=0
TurnstileAlert=0
SecurityBoothAuth=0
SecurityBoothAuthDeny=0
PendingAlarms=0

[Reader1]
CardDecode=30
BaudrateReader=57600
CardLayoutLength=0
CardCodeBegin=0
CardCodeLength=6
EditionBegin=0
EditionLength=0
ShowCardCodeBegin=0
ShowCardCodeLength=0
FacilityCodeBegin=0
FacilityCode=
Direction=2
DirectionInput=0
SkipBioVerify=0
DisableFunctions=0

[Reader2]
CardDecode=99
BaudrateReader=57600
CardLayoutLength=0
CardCodeBegin=0
CardCodeLength=6
EditionBegin=0
EditionLength=0
ShowCardCodeBegin=0
ShowCardCodeLength=0
FacilityCodeBegin=0
FacilityCode=
Direction=2
DirectionInput=0
DisableFunctions=0

[ExtReader]
CardDecode=99
BaudrateReader=57600
CardLayoutLength=0
CardCodeBegin=0
CardCodeLength=6

EditionBegin=0
EditionLength=0
ShowCardCodeBegin=0
ShowCardCodeLength=0
FacilityCodeBegin=0
FacilityCode=
WiegandOutput=0
Direction=2
DirectionInput=0
SkipBioVerify=0
DisableFunctions=0
ReaderLeds=1

[Biometric]
Enabled=0
FreeScan=0
SecurityLevel=16
Sensitivity=8
ImageQuality=2
LightingCondition=0
FastMode=7
MifareFirstBlock=1
TemplateSource=2
FreePass=0
EnrollAuth=0
EnrollAll=0
MinimumQuality=70
SaveReaderSource=1
SendTemplate=1

[System]
TurnOffTimeout=10
TurnoffBacklight=1
Backlight=1
Contrast=5
TurnoffEthernet=0
VirtualKeyIn1=0
VirtualKeyIn2=0
LogLevel=2
LogAggressiveFlush=0
TTY1Config=1
Language=English
FontEncoding=0
SecureOperatorPassword=6D8B9EF12C9B40ECBEC4109BB0479F9B17353E8A64BFB50DA0074803A6FCD304
RemoteUsernameCrypto=4386CF3B44F59C866B7F8006CA5A75B571721A965706259D57F7500171051E19
SecureRemotePassword=4386CF3B44F59C866B7F8006CA5A75B571721A965706259D57F7500171051E19
ManagerUsernameCrypto=CD245F818F8E477B9BD47CC221142F043B65100070BC9F05A6F8C8CF06CBA151
SecureManagerPassword=CD245F818F8E477B9BD47CC221142F043B65100070BC9F05A6F8C8CF06CBA151
TimeLock=
AudioVolume=1
TTY1Legacy=1
TTY1Busy=0
CryptoEnabled=1
TrnsHash=1

FirmwareKey=
Firmware=*VNN*build*nnnn*

[TimeSettings]
SMonth=99
SDay=99
SHour=99
SMin=99
WMonth=99
WDay=99
WHour=99
WMin=99
AutoDayLightSavingTime=1
RecordDayLightSaving=0
UTC=+00:00
RecordUTC=0
UseNTP=0
NTPServerName=pool.ntp.org
NTPRefresh=604800

[Ethernet]
DHCP=1
IPAddress=192.168.1.240
Gateway=0.0.0.0
Subnet=255.255.255.0
Primary_DNS=0.0.0.0
Secondary_DNS=0.0.0.0
FtpPort=21
HttpPort=80
MasterUrl=
Protocol=0
httpOnlineMessage=/online?trsn=$transaction$&id=$termid$&mac=mac
httpBatchMessage=/batch?trsn=$transaction$&id=$termid$&mac=mac
httpKeepAliveMessage=/keepalive?id=$termid$&mac=mac&date=$date$&time=$time$&localtrsn=$localtransaction$
TermID=”X1”
ConnTimeout=5
KeepAliveInterval=15
RetryTimeout=60
EnableHTTPServer=1
EnableFTPServer=1
EncodeUrl=0

[GPRS]
Enabled=0
ConnectionInterval=0
ATextraCommand=""
Dialnum=""
User=""
Password=""

[FtpClient]
ServerURL=
User=""
Password=""

PassiveMode=1
RetryNumber=3
ResponseTimeout=10
DestinationFileName="TRANSACTIONS.TXT"
FileOpeningMode=0

[USB]
Enabled=0
SecurePasswordUSB=6D8B9EF12C9B40ECBEC4109BB0479F9B17353E8A64BFB50DA0074803A6FCD304
TrnsFileUSB=TRANSACTIONS.TXT
MoveTrnsToUSB=1

[Printer]
Enabled=0
Baudrate=9600

[Aperio]*
KeyA=*xxxxxxxxxxxxxxxx*
KeyB=*xxxxxxxxxxxxxxxx*
SiteCode=*xxxxxxxxxxxxxxxx*
Integrity=*xxxxxxxxxxxxxxxx*
StandardSector=0
AccessMatrix=00000000
AlarmsSector=00000000

Nota: la sezione marcata con un asterisco (*) è presente solamente nella versione speciale di firmware Aperio (vedi §18 a pag. 165).

4.12 ATTIVAZIONE DI FUNZIONI OPZIONALI DEL FIRMWARE

Il parametro **FirmwareKey** nella sezione *[System]* del file PARAMETERS.TXT consente di sbloccare, previa richiesta di un'apposita chiave di attivazione a Zucchetti AXESS, uno o più moduli opzionali del firmware.

Per richiedere una chiave di attivazione è necessario specificare le seguenti informazioni per ogni terminale di cui si vogliono estendere le funzionalità:

1) Codice prodotto (o *part number*, p/n)
2) Numero di serie (o *serial number*, s/n)
3) Identificatore unico del terminale (o *ID*)
4) Lista dei moduli firmware opzionali da attivare

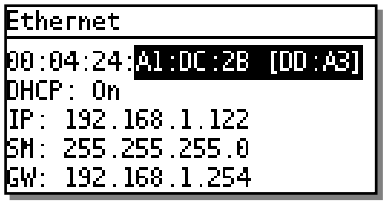
Codice prodotto e numero di serie

Il codice prodotto (o *part number*, p/n) ed il numero di serie (o *serial number*, s/n) sono entrambi stampati sull'etichetta identificativa del prodotto, di cui si possono trovare 2 copie: una (più piccola) è attaccata sulla parte posteriore dell'involucro del terminale, mentre l'altra (più grande) si trova sulla scatola di cartone con cui il terminale è stato consegnato.

Identificatore unico del terminale

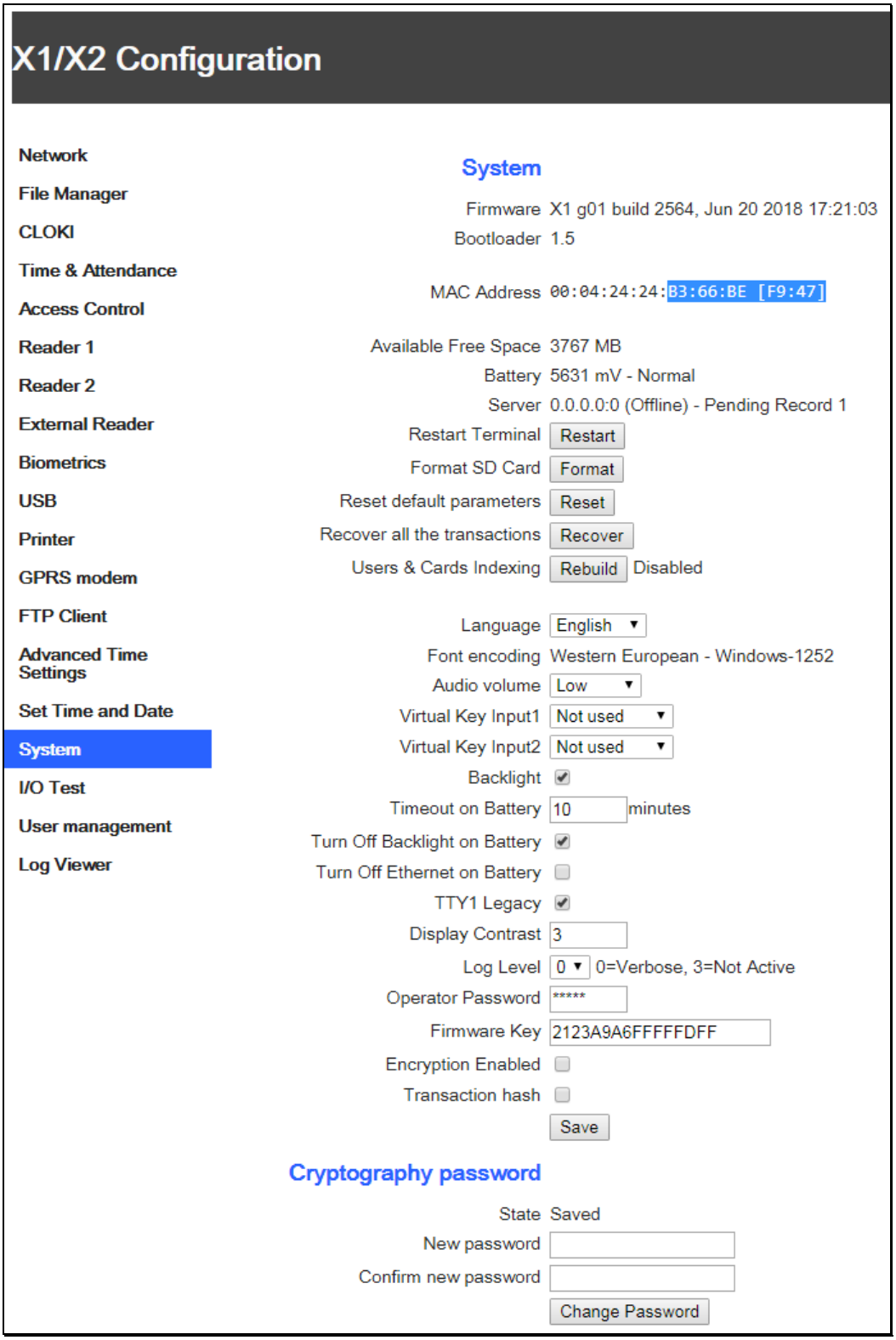
L'identificatore unico del terminale (o *ID*) è una stringa di 10 cifre esadecimali memorizzata all'interno del terminale. Vi si può risalire in 3 modi diversi:

- All'interno del menu supervisore (vedi §10.5 a pag. 113), e precisamente nel sottomenu "**Info/Ethernet**", compare una stringa di 12+4 cifre esadecimali, di cui le prime 12 rappresentano l'indirizzo MAC, mentre le ultime 10 rappresentano l'identificatore unico richiesto (nota: i caratteri di separazione non sono rilevanti):



Le stesse informazioni appaiono anche temporaneamente (per circa 3 secondi) nella seconda schermata mostrata all'avvio del terminale (vedi §10.1 a pag. 108).

- Nella pagina "**System**" del web server http del terminale compare la stessa stringa descritta al punto precedente:



- Inviando il comando Ethernet di basso livello **h** (pacchetto di tipo "6" nel protocollo TMC-UDP) alla porta UDP 8499 del terminale, il quale risponderà con un pacchetto dello stesso tipo contenente la stessa stringa descritta ai punti precedenti, ad eccezione dei caratteri di separazione che come detto non sono rilevanti, ad es. 00:04:24:**B3:66:BE:F9:47**

Lista dei moduli firmware opzionali da attivare

- X1/X2 GATE MANAGER

  Consente la gestione avanzata di un varco di controllo accessi. **Note**: 1) a partire dalla versione di firmware "a12_build802" questo modulo è sempre attivo (non richiede più una chiave di attivazione); 2) su versioni di firmware precedenti, se X1/X2 viene gestito dal programma Xatl@s, la chiave di attivazione per la gestione avanzata del varco viene caricata automaticamente, quindi non è necessario farne richiesta né inserirla manualmente.

- X1/X2 RFID2 SERIAL ZUCCHETTI

  Consente la decodifica corretta delle carte Mifare Zucchetti e la visualizzazione a display del nome utente contenuto nella carta.

- X1/X2 APERIO ENCODER

  Consente la creazione delle ACCESS CARDS da usare in un impianto di serrature wireless Aperio offline (vedi §18 a pag. 165), partendo da carte Mifare vuote.

INSERIMENTO DELLA CHIAVE DI ATTIVAZIONE

Ogni chiave di attivazione è una stringa di 16 cifre esadecimali (ad es. 20EB0238FFFFFEFF), ed è valida solo per il terminale il cui identificatore unico è stato usato per generarla. Per caricare la chiave di attivazione è sufficiente impostare il parametro **FirmwareKey** nella sezione *[System]* del file PARAMETERS.TXT oppure, analogamente, inserire la chiave nell'apposita casella di testo "**Firmware Key**" nella pagina "**System**" del web server HTTP (vedi figura qui sopra).

Attenzione: la chiave di attivazione viene mantenuta anche nel caso in cui venga effettuata l'operazione "**Format SD Card**" dalla pagina "**System**" del web server HTTP, o comunque in caso di cancellazione del file PARAMETERS.TXT. Invece, effettuando la formattazione della SD card da PC, questa impostazione viene rimossa.

5. TABELLE DI CONTROLLO ACCESSI

Impostando a 1 il parametro **Enable** nella sezione *[AccessControl]* del file PARAMETERS.TXT (vedi §4.11 a pag. 36) oppure, analogamente, spuntando la checkbox "**Enable**" nella pagina "**Access Control**" del web server HTTP, si attiva la funzionalità di controllo accessi.

La logica di controllo accessi viene effettuata autonomamente dal terminale in seguito alla lettura di una tessera, in base ai criteri definiti da una serie di file di sistema. Questi file (8 in tutto, di cui 1 sempre necessario, più altri 3 necessari solo in caso di definizione delle fasce orarie e ulteriori 4 opzionali) hanno un formato specifico descritto nei paragrafi successivi.

Avvertenza: se X1/X2 viene gestito dal programma Xatl@s, i file necessari vengono caricati automaticamente e non devono essere modificati, inoltre alcune delle informazioni fornite in questo capitolo sono valide solo nel caso in cui Xatl@s non sia utilizzato.

All'interno di ciascun file, ogni record viene identificato da un codice univoco (chiave primaria) che ne costituisce la parte iniziale e viene utilizzato per realizzare riferimenti incrociati fra le differenti tipologie di dati sino ad ottenere le informazioni complete. Ad esempio, nel record che definisce una tessera abilitata c'è un riferimento all'ID del record che definisce i dati anagrafici degli utenti. In tal modo, dal codice rilevato al momento della transazione, è possibile risalire al nominativo associato alla tessera e visualizzarlo sul display del terminale. Nei successivi paragrafi, i riferimenti incrociati fra i file sono evidenziati usando lo stesso colore dei caratteri per i campi dati che hanno lo stesso significato in file diversi.

5.1 FILE NECESSARI PER IL CONTROLLO ACCESSI

L'unico file che è sempre necessario quando il controllo accessi è abilitato è il seguente:

CARDS.TXT

Contiene la lista dei codici delle tessere riconosciute dal sistema. Ogni codice tessera può essere associato ad un gruppo di autorizzazioni per stabilire le regole di accesso (in caso di definizione delle fasce orarie) e, opzionalmente, ad un codice univoco associato all'utente (il quale teoricamente può essere associato a più di una tessera abilitata) che consente di visualizzarne il nome sul display del terminale o di richiedere l'introduzione di un codice PIN per la conferma del passaggio della tessera. Inoltre, mediante un apposito *flag*, ogni codice tessera può essere abilitato o disabilitato a priori, indipendentemente dalla validità del gruppo di autorizzazioni (utile per usare CARDS.TXT come ***black list***, anche solo parzialmente, vedi sotto).

oppure, in alternativa:

CARDRNGE.TXT

Oltre alla gestione delle singole tessere abilitate all'accesso, è possibile definire una serie di intervalli all'interno dei quali ogni codice tessera viene considerato valido. I record di questo file, come quelli contenuti in CARDS.TXT, permettono di definire i gruppi di autorizzazione per stabilire le regole di accesso (in caso di definizione delle fasce orarie). In questo caso, tuttavia, non è possibile associare i codici tessera all'anagrafica degli utenti e neppure utilizzare i PIN, in quanto le tessere vengono trattate come gruppi e non singolarmente, non definendo il codice univoco associato a ciascun utente.

CARDS.TXT e CARDRNGE.TXT possono comunque essere presenti entrambi contemporaneamente senza nessuna controindicazione. In questo caso, se un codice tessera è definito in CARDS.TXT e fa anche parte di un intervallo definito in CARDRNGE.TXT, le regole stabilite da CARDS.TXT hanno priorità per la gestione dell'accesso. Ad esempio, se CARDS.TXT contiene un codice non abilitato e CARDRNGE.TXT contiene un intervallo valido in cui è contenuto tale codice, l'accesso viene comunque negato: questo sistema può essere sfruttato per usare CARDS.TXT come ***black list*** , anche solo parzialmente. Se

invece un codice tessera non è definito in CARDS.TXT ma fa parte di un intervallo definito in CARDRNGE.TXT, per la gestione dell'accesso viene preso in considerazione solo quest'ultimo, come se CARDS.TXT non ci fosse.

Nel caso in cui non vengano definite le fasce orarie e tutti i codici tessera abbiano gli stessi diritti di accesso, non è strettamente necessario caricare altri file per gestire gli accessi (a meno che non si desideri visualizzare i nomi degli utenti o gestire l'introduzione del PIN, nel qual caso è necessario almeno anche il file **USERS.TXT**, vedi §5.2 a pag. 69). In caso contrario, devono essere caricati almeno altri 3 file (nota: o non si carica nessuno dei seguenti file o si caricano tutti e 3, altrimenti si incorre sempre in una condizione di errore):

AUTHGRP.TXT

Contiene la definizione dei gruppi di autorizzazioni. Un gruppo di autorizzazioni è un insieme di un numero fisso (32) di autorizzazioni, che a loro volta consentono di definire le fasce orarie che regolano l'accesso. Utilizzando i gruppi di autorizzazioni è possibile associare più codici tessera ad un unico modello comportamentale, ad esempio consentendo l'accesso a tutti i dipendenti usando la stessa tabella oraria.

AUTH.TXT

Contiene la definizione delle singole autorizzazioni. Dal punto di vista logico, utilizzando le autorizzazioni è possibile abilitare l'accesso di un determinato codice tessera in fasce orarie diverse su terminali diversi. In realtà X1 e X2 gestiscono solo gli accessi sul terminale di console (non possono funzionare come "*master*" di controllo accessi regolando i transiti su altre unità "*slave*"), quindi ogni autorizzazione permette solo di associare un codice tessera ad un numero fisso (8) di modelli orari, consentendo quindi l'accesso all'interno di un certo numero di fasce orarie limitate.

TIMEMOD.TXT

Contiene la definizione dei modelli orari in cui consentire l'accesso. Un modello orario è un insieme di un numero fisso (24) di fasce orarie differenti, ciascuna delimitata da un orario iniziale ed uno finale, attivabili in base al giorno della settimana. Tramite un file opzionale (**CALENDAR.TXT**, vedi §5.2 qui sotto) si può definire un calendario delle festività personalizzato e attivare o meno una fascia oraria anche nei giorni festivi.

5.2 FILE OPZIONALI PER IL CONTROLLO ACCESSI

I file elencati al precedente §5.1 sono necessari per il corretto funzionamento delle funzioni di base del controllo accessi. Se lo si desidera, mediante alcuni file opzionali e logicamente collegati ai precedenti, è possibile attivare le funzionalità avanzate del controllo accessi:

USERS.TXT

Contiene l'anagrafica degli utenti. Ogni record è identificato da un codice utente univoco, a cui è possibile fare riferimento nei record contenuti nel file **CARDS.TXT** (vedi §5.1 a pag. 68). Se questo collegamento viene attivato, al momento del passaggio della tessera nella schermata principale del terminale comparirà il nome dell'utente invece del codice della tessera. Tramite questo file è inoltre possibile definire tipologie di utenti per gestire causali personalizzate (a tale scopo è necessario caricare anche il file opzionale **AXREASON.TXT**, vedi sotto). Infine, USERS.TXT consente anche di associare un codice PIN ad un utente: in questo caso si potrà poi decidere se le transazioni debbano essere accettate subito dopo il passaggio della tessera oppure solo dopo avere richiesto l'inserimento del PIN ed averne verificato la correttezza. Nota: esiste anche una modalità di funzionamento denominata "solo PIN" che consente di effettuare l'accesso semplicemente digitando il codice PIN associato all'utente abilitato, senza richiedere la presenza di una tessera fisica (solo sui modelli X2 con tastiera numerica, vedi §5.12 a pag. 81).

AXREASON.TXT

Contiene l'elenco delle causali speciali di ingresso / uscita per la rilevazione delle presenze. La differenza principale fra questo file ed il file **REASONS.TXT** descritto al §4.4 a pag. 19 è che in questo caso ciascuna causale è associata ad un numero fisso (4) di tipologie di utenti (definite nell'altro file opzionale **USERS.TXT**, vedi sopra), quindi non è necessariamente valida per tutti gli utenti. Utilizzando questo file, quindi, è possibile definire dei codici associati a causali personalizzate, selezionabili dall'operatore direttamente sul terminale nello stesso modo in cui vengono selezionate le causali contenute in REASONS.TXT (tasto ▼). Il terminale mostra comunque le descrizioni di tutte le causali contenute in AXREASONS.TXT, ma una volta selezionata una causale ed effettuata la lettura della tessera, la transazione sarà accettata solo se il codice tessera corrisponde ad una tipologia di utente consentita per quella causale. Nota: AXREASON.TXT e REASONS.TXT possono comunque essere presenti entrambi contemporaneamente senza controindicazioni: in questo caso però viene preso in considerazione solo AXREASON.TXT, come se REASONS.TXT non ci fosse. Se è presente solo REASONS.TXT, le causali selezionate vengono sempre considerate valide, indipendentemente dall'utente. Se invece nessuno di questi file è presente, non è possibile specificare alcuna causale al momento della transazione.

CALENDAR.TXT

Contiene il calendario delle festività personalizzate. Tramite questo file è possibile definire un numero fisso di date (96) nelle quali può essere attivata una certa fascia oraria, indipendentemente dal giorno della settimana in cui cade tale data. In caso il file non sia presente, le fasce orarie determineranno il funzionamento del controllo accessi esclusivamente in base al giorno della settimana, senza tenere conto della data corrente. In pratica, il file **CALENDAR.TXT** ha per **TIMEMOD.TXT** (vedi §5.1 a pag. 68) lo stesso significato che il file **HOLIDAYS.TXT** (§4.3 a pag. 19) ha per **ALARMS.TXT** (§4.2 a pag. 18), ma mentre i primi due vengono usati esclusivamente per il controllo accessi, gli ultimi due vengono usati esclusivamente per l'attivazione temporizzata dei relé.

5.3 FORMATO DEI FILE PER IL CONTROLLO ACCESSI

Ciascuno dei file di controllo accessi elencati ai §5.1 e §5.2 ha un diverso formato, ottimizzato in base alla tipologia dei dati che contiene. Sono però individuabili alcune caratteristiche comuni a tutti i file:

1. All'interno dei file sono ammessi solo caratteri ASCII. Alcuni caratteri sono riservati per la corretta interpretazione dei dati e quindi non devono essere usati se non ove espressamente specificato:
   - il carattere underscore '_' (5Fh) viene usato come separatore fra i campi all'interno di un record
   - i campi contenenti solo caratteri '0' (30h) vengono utilizzati per inserire un dato non significativo, permettendo comunque di mantenere la corretta lunghezza del record. Attenzione però: non è possibile utilizzare un campo del tipo "000…000" come identificatore univoco di un record (cioè la cosiddetta "chiave primaria", che normalmente corrisponde al primo campo di ogni record). Il carattere '0' può essere comunque usato all'interno di dati significativi per rappresentare la cifra "zero" e, in alcuni casi, come flag per disabilitare un record senza necessariamente rimuoverlo o invalidarlo
   - la coppia di byte <CR><LF> (0Dh 0Ah), che indicheremo nel seguito CR LF per brevità, viene utilizzata come separatore fra i record all'interno dei file che consentono la registrazione di record multipli.
   - il carattere '$' viene usato per invalidare un record senza rimuoverlo (il che significherebbe ricaricare l'intero file)
2. Tutti i file, tranne CALENDAR.TXT (vedi 5.11 a pag. 80), possono contenere un numero variabile di record. Però i record hanno sempre una lunghezza fissa, e devono tutti terminare con i caratteri CR LF, compreso l'ultimo. Ne consegue che, anche se la dimensione totale dei file è variabile, questa è sempre divisibile per la lunghezza del record ivi contenuto, e gli ultimi due byte in ogni file devono sempre essere CR LF (ne consegue che il file deve sempre terminare con una linea vuota)
3. Il nome dei file deve sempre essere maiuscolo, e tutti i file utilizzati devono trovarsi nella *root* principale della scheda di memoria micro-SD del terminale

La gestione del controllo accessi deve essere abilitata, come descritto al §5 a pag. 68. Se il controllo accessi viene disabilitato, i file già presenti nel terminale vengono mantenuti, ma non viene più effettuato alcun controllo. Se invece viene abilitato il controllo accessi in mancanza di uno qualunque dei file richiesti, ogni tentativo di transazione viene rifiutato. Analogamente, se si verifica un errore nell'interpretazione del contenuto dei file (ad esempio un errore nella formattazione di un record in un file qualsiasi), ogni tentativo di transazione viene rifiutato.

Si ricordi inoltre che se in un record vi è un campo che fa riferimento alla chiave primaria di un record in un altro file (come ad esempio se in un record di CARDS.TXT viene effettuato il riferimento all'anagrafica contenuta in USERS.TXT), tale collegamento deve essere risolto correttamente. Nel caso in cui il record a cui si fa riferimento non esista, o non sia riconosciuto valido, la transazione sarà rifiutata.

5.4 CARDS.TXT

Contiene l'elenco dei codici tessera riconosciuti dal sistema. Ogni record può avere una lunghezza fissa di 69 byte (cioè 67 caratteri + CR LF) oppure 71 byte(cioè 69 caratteri + CR LF), e permette di associare un codice tessera all'identificativo di un gruppo di autorizzazioni definito nel file AUTHGRP.TXT (vedi §5.6 a pag. 74). Opzionalmente si può definire un periodo di validità della tessera e associarne il codice anche all'identificativo di un utente definito nel file USERS.TXT (vedi §5.9 a pag. 77). Ogni record è interpretato secondo uno dei 2 possibili formati seguenti (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. 70):

R_CCCCCCCCCCCCCCCC_GGGGGGGGGG_IIIIIIIIII_AAMMGGHHMM_aammgghhmm_E_ee CR LF

oppure

R_CCCCCCCCCCCCCCCC_GGGGGGGGGG_IIIIIIIIII_AAMMGGHHMM_aammgghhmm_E_ee_B CR LF

a seconda che si desideri utilizzare anche il flag opzionale **B** (biometrico) oppure no: in ogni caso tutti i record di CARDS.TXT devono avere la stessa lunghezza (rispettivamente 71 o 69 byte). La differenza è la seguente: se si utilizza la versione senza flag biometrico, tutti i codici tessera elencati in CARDS.TXT possono essere usati per la registrazione di impronte su un eventuale modulo biometrico esterno FingerBOX (vedi §11 a pag. 120); se si utilizza la versione con flag biometrico, invece, e se il parametro **EnrollAuth**=1 nella sezione *[Biometric]* del file PARAMETERS.TXT (vedi §4.11 a pag. 49), solo i codici tessera autorizzati mediante l'impostazione del flag a '1' possono essere usati per la registrazione di impronte.

Vediamo il significato dei vari campi:

R — 1 byte che identifica il tipo di lettore da cui è consentito ricevere il codice tessera (**nota**: se il file CARDS.TXT è presente, questo controllo viene effettuato anche durante la registrazione di un'impronta, a prescindere dal fatto che il controllo accessi sia attivato oppure no: se il lettore usato non corrisponde al valore di questo campo, l'operazione di *enrollment* viene abortita con il messaggio di errore "**Operazione fallita – Tessera non valida**").

0: provenienza della lettura indifferente

1: il codice tessera deve provenire dal lettore primario (READER 1) o inserimento manuale

2: il codice tessera deve provenire dal lettore secondario (READER 2) (solo in assenza di un modulo biometrico esterno FingerBOX)

3: la lettura deve provenire dal lettore esterno su morsettiera a vite (EXTERNAL READER) o da eventuali lettori aggiuntivi collegati su schede di espansione 914 NeoMAX opzionali, vedi §3.6 a pag. 11.

Per le eventuali transazioni effettuate mediante un modulo biometrico esterno FingerBOX in modalità "solo dito" (identificazione 1:N), cioè senza previa lettura/inserimento di un codice tessera, questo campo viene confrontato con l'analogo campo **R** nel record relativo allo stesso codice tessera all'interno del file USERCODS.TXT (vedi §11.1 a pag. 127).

Nota: l'utente è ora strettamente legato non solo al codice tessera, ma anche al lettore utilizzato per effettuare la lettura della tessera. E' quindi possibile associare 2 record con

lo stesso codice tessera ma provenienza della lettura diversa ad identificatori di utenti diversi.

CCCCCCCCCCCCCCCC — 16 byte che rappresentano il codice univoco della tessera (chiave primaria). Questo dato è richiesto per ogni record e deve essere univoco all'interno del file e diverso da "000…000". Data la necessità di mantenere la lunghezza fissa del record, se il codice letto è più corto di 16 cifre è comunque necessario anteporre alla parte significativa del codice tanti caratteri '0' di riempimento fino a raggiungere i 16 byte.

GGGGGGGGGG — 10 byte che rappresentano l'identificatore univoco del gruppo di autorizzazioni associato alla tessera (definito nel file AUTHGRP.TXT, vedi §5.6 a pag. 74). Questo dato è facoltativo e può essere semplicemente riempito con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record, ma solo se i file AUTHGRP.TXT / AUTH.TXT / TIMEMOD.TXT non sono presenti (comunque in questo caso il valore del campo è inifluente).

IIIIIIIIII — 10 byte che rappresentano l'identificatore univoco dell'utente associato alla tessera (definito nel file USERS.TXT, vedi §5.9 a pag. 77). Questo dato è facoltativo e può essere semplicemente riempito con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record. In caso il file USERS.TXT sia presente, l'utente associato sia attivo ed i suoi dati siano completi, questi ultimi verranno mostrati nella schermata principale del terminale in seguito al passaggio della tessera.

AAMMGGHHMM — 10 byte che rappresentano la data e ora di inizio validità della tessera. Si devono indicare le 2 ultime cifre dell'anno, 2 cifre per il mese, 2 cifre per il giorno, 2 cifre per l'ora e 2 cifre per i minuti (ad esempio, volendo inserire le 10:30 del 26 settembre 2011, sarà necessario indicare 1109261030). Nota: l'orario 00:00 corrisponde sempre all'inizio della giornata specificata, mentre 23:59 corrisponde alla fine della giornata (valore massimo utilizzabile). Questo dato è facoltativo e può essere semplicemente riempito con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record; in caso contrario, se la tessera transita in un periodo precedente alla data specificata, essa viene rifiutata.

aammgghhmm — 10 byte che rappresentano la data e ora di fine validità della tessera, espresse usando lo stesso formato della data di inizio validità. Nota: l'orario 00:00 corrisponde sempre all'inizio della giornata specificata, mentre 23:59 corrisponde alla fine della giornata (valore massimo utilizzabile). Questo dato è facoltativo e può essere semplicemente riempito con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record; in caso contrario, se la tessera transita in un periodo successivo alla data specificata, essa viene rifiutata.

E — 1 byte che rappresenta il flag di abilitazione della tessera. '1' significa tessera abilitata all'accesso, '0' significa tessera definita in archivio ma attualmente disabilitata, cioè transito non consentito (utile per usare CARDS.TXT come ***black list***, anche solo parzialmente).

ee — 2 byte che rappresentano il codice edizione della tessera. Il controllo del codice edizione viene effettuato solo se il parametro **EditionLength** (vedi §4.7 a pag. 45) nella sezione PARAMETERS.TXT relativa al lettore usato per leggere la carta è diverso da '0': in tutti gli altri casi questo campo non viene considerato, ma deve comunque essere riempito con 2 caratteri qualunque (ad esempio "00") per mantenere la lunghezza fissa del record. Se **EditionLength**=1, alla il codice edizione a singola cifra viene confrontato con il secondo carattere (quello più a destra) di questo campo; il primo (quello più a sinistra) non viene considerato e può avere qualunque valore (ad esempio "0"). Se **EditionLength** è diverso da '0', impostare questo campo a "00" significa che il codice edizione letto deve essere '0' o '00', altrimenti la transazione non verrà accettata.

B 1 byte (opzionale) che rappresenta il flag di autorizzazione alla registrazione di impronte su un eventuale modulo biometrico esterno FingerBOX (vedi §11 a pag. 120). '1' significa tessera autorizzata all'utilizzo della biometria, '0' significa tessera non autorizzata. ***Nota***: ha effetto solo se il parametro **EnrollAuth**=1 nella sezione *[Biometric]* del file PARAMETERS.TXT (vedi §4.11 a pag. 49).

Esempio di record di CARDS.TXT:

0_0000000000004269_0000000003_0000000001_1102011830_0000000000_1_00CRLF

Definisce la tessera con codice 4269, che utilizza l'eventuale gruppo di autorizzazioni con identificatore 3 (nel caso in cui vengano caricati i file AUTHGRP.TXT, AUTH.TXT e TIMEMOD.TXT), associata all'eventuale utente con identificatore 1 (nel caso in cui venga caricato il file USERS.TXT). La tessera è valida a partire dalle 08:30 del 1 febbraio 2011, senza alcuna scadenza, abilitata all'accesso e senza controllo dell'edizione. In questo caso il flag biometrico non è stato usato.

5.5 CARDRNGE.TXT

Contiene l'elenco degli intervalli di codici tessera abilitati all'accesso. Ogni record ha una lunghezza fissa di 81 byte (cioè 79 caratteri + CR LF) e permette di associare un intervallo di codici tessera all'identificativo di un gruppo di autorizzazioni definito nel file AUTHGRP.TXT (vedi §5.6 a pag. 74), secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. 70):

RRRRRRRRRR_CCCCCCCCCCCCCCCC_CCCCCCCCCCCCCCCC_GGGGGGGGGG_AAMMGGHHMM_aammgghhmm_ECR LF

Dove:

RRRRRRRRRR 10 byte che rappresentano l'identificatore univoco dell'intervallo di codici tessera (chiave primaria). Questo dato è richiesto per ogni record e deve essere univoco all'interno del file e diverso da "000…000".

CCCCCCCCCCCCCCCC 16 byte che rappresentano il limite inferiore dell'intervallo di codici tessera abilitati. Data la necessità di mantenere la lunghezza fissa del record, se il limite inferiore è più corto di 16 cifre è comunque necessario anteporre alla parte significativa del codice tanti caratteri '0' di riempimento fino a raggiungere i 16 byte.

CCCCCCCCCCCCCCCC 16 byte che rappresentano il limite superiore dell'intervallo di codici tessera abilitati. Data la necessità di mantenere la lunghezza fissa del record, se il limite superiore è più corto di 16 cifre è comunque necessario anteporre alla parte significativa del codice tanti caratteri '0' di riempimento fino a raggiungere i 16 byte.

GGGGGGGGGG 10 byte che rappresentano l'identificatore univoco del gruppo di autorizzazioni associato a tutte le tessere comprese nell'intervallo (definito nel file AUTHGRP.TXT, vedi §5.6 a pag. 74). Questo dato è facoltativo e può essere semplicemente riempito con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record, ma solo se i file AUTHGRP.TXT / AUTH.TXT / TIMEMOD.TXT non sono presenti (comunque in questo caso il valore del campo è inifluente).

AAMMGGHHMM 10 byte che rappresentano la data e ora di inizio validità di tutte le tessere comprese nell'intervallo. E' equivalente all'analogo campo di CARDS.TXT (vedi §5.4 a pag. 71) ed ha lo stesso formato. Nota: l'orario 00:00 corrisponde sempre all'inizio della giornata specificata, mentre 23:59 corrisponde alla fine della giornata (valore massimo utilizzabile). Questo dato è facoltativo e può essere semplicemente riempito con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record; in caso contrario, se una tessera

compresa nell'intervallo transita in un periodo precedente alla data specificata, essa viene rifiutata.

aammgghhmm — 10 byte che rappresentano la data e ora di fine validità di tutte le tessere comprese nell'intervallo. E' equivalente all'analogo campo di CARDS.TXT (vedi §5.4 a pag. 71) ed ha lo stesso formato. Nota: l'orario 00:00 corrisponde sempre all'inizio della giornata specificata, mentre 23:59 corrisponde alla fine della giornata (valore massimo utilizzabile). Questo dato è facoltativo e può essere semplicemente riempito con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record; in caso contrario, se una tessera compresa nell'intervallo transita in un periodo successivo alla data specificata, essa viene rifiutata.

E — 1 byte che rappresenta il flag di abilitazione dell'intervallo di codici tessera. '1' significa che tutte le tessere con codici compresi nei limiti definiti per l'intervallo sono abilitate all'accesso, '0' significa intervallo definito in archivio ma attualmente disabilitato (transito non consentito a tutte le tessere comprese nell'intervallo).

Esempio di record di CARDRNGE.TXT:

0000000001_0000000000000042_0000000000000150_0000000003_1103050830_1206131730_1CRLF

Definisce un intervallo con identificatore 1 per i codici tessera che vanno da 42 fino a 150, che utilizza l'eventuale gruppo di autorizzazioni con identificatore 3 (nel caso in cui vengano caricati i file AUTHGRP.TXT, AUTH.TXT e TIMEMOD.TXT), valido dalle 08:30 del 5 marzo 2011 fino alle 17:30 del 13 giugno 2012, ed abilitato all'accesso.

5.6 AUTHGRP.TXT

Contiene la definizione dei gruppi di autorizzazioni. Ogni record ha una lunghezza fissa di 364 byte (cioè 362 caratteri + CR LF) e permette di associare ad ogni identificativo di gruppo fino a 32 singole autorizzazioni definite nel file AUTH.TXT (vedi §5.7 a pag. 75), secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. 70):

GGGGGGGGGG_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_
IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_
IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_
IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_
IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIII_IIIIIIIIIICRLF

Dove:

GGGGGGGGGG — 10 byte che rappresentano l'identificatore univoco del gruppo di autorizzazioni (chiave primaria) a cui viene fatto riferimento nei file CARDS.TXT (vedi §5.4 a pag. 71) e/o CARDRNGE.TXT (vedi §5.5 a pag. 73). Questo dato è richiesto per ogni record e deve essere univoco all'interno del file e diverso da "000…000".

IIIIIIIIII — 10 byte che rappresentano l'identificatore univoco dell'autorizzazione (definito nel file AUTH.TXT (vedi §5.7 a pag. 75). Ad ogni record di AUTHGRP.TXT si possono associare fino a 32 campi di questo tipo, anche se solo il primo deve necessariamente essere definito. Se è sufficiente associare una sola autorizzazione al gruppo, è comunque necessario riempire tutti i campi seguenti con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record.

Esempio di record di AUTHGRP.TXT:

0000000003_0000000004_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0

000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000
000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000_0000000000CRLF

Definisce un gruppo di autorizzazioni con identificatore 3, che utilizza solo l'autorizzazione con identificatore 4.

5.7 AUTH.TXT

Contiene la definizione delle singole autorizzazioni. Ogni record ha una lunghezza fissa di 111 byte (cioè 109 caratteri + CR LF) e permette di associare ad un determinato terminale (in questo caso sempre lo stesso, poiché X1 e X2 non possono funzionare come "*master*" di controllo accessi regolando i transiti su altre unità "*slave*") fino ad 8 modelli orari definiti nel file TIMEMOD.TXT (vedi §5.8 a pag. 75), secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. 70):

IIIIIIIIII_0000000001_MMMMMMMMMM_MMMMMMMMMM_MMMMMMMMMM_MMMMMMMMMM_MMMMMMMMMM_
MMMMMMMMMM_MMMMMMMMMM_MMMMMMMMMMCR LF

Dove:

IIIIIIIIII	10 byte che rappresentano l'identificatore univoco dell'autorizzazione (chiave primaria) a cui viene fatto riferimento nel file AUTHGRP.TXT (vedi §5.6 a pag. 74). Questo dato è richiesto per ogni record e deve essere univoco all'interno del file e diverso da "000…000".
0000000001	Stringa fissa di 10 byte che rappresenta l'identificatore dell'unico terminale gestito dall'applicazione di controllo accessi.
MMMMMMMMMM	10 byte che rappresentano l'identificatore univoco del modello orario di riferimento (definito nel file TIMEMOD.TXT, vedi §5.8 a pag. 75). Ad ogni record di AUTH.TXT si possono associare fino a 8 campi di questo tipo, anche se solo il primo deve necessariamente essere definito. Se è sufficiente associare un solo modello orario all'autorizzazione, è comunque necessario riempire tutti i campi seguenti con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record.

Esempio di record di AUTH.TXT:

0000000004_0000000001_0000000005_0000000000_0000000000_0000000000_0000000000_0000000000_000000
0000_0000000000CRLF

Definisce un'autorizzazione con identificatore 4, associata solo al modello orario con identificatore 5.

5.8 TIMEMOD.TXT

Contiene la definizione delle fasce orarie in cui consentire l'accesso. Ogni record può avere una lunghezza fissa di 468 byte (cioè 466 caratteri + CR LF) oppure 470 byte (cioè 468 caratteri + CR LF), e permette di associare a ciascun identificativo di modello orario fino a 24 fasce orarie, secondo uno dei 2 possibili formati seguenti (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. 70):

MMMMMMMMMM_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSFCR LF

oppure

MMMMMMMMMM_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_HHMM_hhmm_DLMMGVSF_
HHMM_hhmm_DLMMGVSF_dCRLF

a seconda che si desideri utilizzare anche il flag opzionale **d** (direzione) oppure no: in ogni caso tutti i record di TIMEMOD.TXT devono avere la stessa lunghezza (rispettivamente 470 o 468 byte). La differenza è la seguente: se si utilizza la versione senza flag di direzione, l'accesso è sempre consentito nelle fasce orarie specificate, indipendentemente dalla direzione di transito; se si utilizza la versione con flag di direzione, invece, solo le transazioni effettuate nella direzione corrispondente al valore del flag verranno accettate.

Vediamo il significato dei vari campi:

MMMMMMMMMM	10 byte che rappresentano l'identificatore univoco del modello orario (chiave primaria) a cui viene fatto riferimento nel file AUTH.TXT (vedi §5.7 a pag. 75). Questo dato è richiesto per ogni record e deve essere univoco all'interno del file e diverso da "000…000".
HHMM	4 byte che rappresentano l'orario iniziale di una fascia oraria (2 cifre per l'ora e 2 cifre per i minuti). Nota: l'orario 00:00 corrisponde sempre all'inizio della giornata, mentre 23:59 corrisponde alla fine della giornata (valore massimo utilizzabile).
hhmm	4 byte che rappresentano l'orario finale di una fascia oraria (nello stesso formato dell'orario iniziale). Nota: l'orario 00:00 corrisponde sempre all'inizio della giornata, mentre 23:59 corrisponde alla fine della giornata (valore massimo utilizzabile).
DLMMGVSF	8 byte che rappresentano i giorni di validità della fascia oraria. Ognuno di questi caratteri, se diverso da '0', abilita l'accesso nel relativo giorno della settimana, mentre inserendo '0' in una qualunque posizione l'accesso nel giorno corrispondente viene disabilitato, indipendentemente dall'orario. In particolare, la prima posizione (D) imposta l'accesso alla domenica, la seconda (L) il lunedì, la terza (M) il martedì e cosi via fino alla settima (S) per il sabato, mentre l'ottava (F) abilita (o disabilita) l'accesso nei giorni festivi, indipendentemente da quale sia il giorno della settimana in cui cadono (nota: l'impostazione relativa ai giorni festivi è sempre più prioritaria rispetto a quelle relative ai giorni della settimana). I giorni festivi vengono definiti tramite il file opzionale CALENDAR.TXT (vedi §5.11 a pag. 80).

L'insieme degli ultimi 3 campi sopra descritti, con i relativi caratteri '_' usati come separatori, rappresenta la definizione completa di una fascia oraria: **HHMM_hhmm_DLMMGVSF**

Ad ogni record di TIMEMOD.TXT si possono associare fino a 24 terzine di campi di questo tipo, anche se solo la prima deve necessariamente essere definita. Se è sufficiente associare una sola fascia oraria al modello, è comunque necessario riempire tutte le terzine di campi seguenti con "0000_0000_00000000" per mantenere la lunghezza fissa del record.

d	1 byte (opzionale) che rappresenta il flag di direzione associato al modello orario. '0' significa che l'accesso è consentito sia per le transazioni in entrata che per quelle in uscita, '1' che vengono accettate solo le transazioni in entrata, '2' che vengono accettate solo quelle in uscita.

Esempio di record di TIMEMOD.TXT:

0000000005_0800_1300_0LMMGV00_1600_1800_0LMMGV00_0000_0000_00000000_0000_0000_00000000_0000_00
00_00000000_0000_0000_00000000_0000_0000_00000000_0000_0000_00000000_0000_0000_00000000_0000_0
000_00000000_0000_0000_00000000_0000_0000_00000000_0000_0000_00000000_0000_0000_00000000_0000_
0000_00000000_0000_0000_00000000_0000_0000_00000000_0000_0000_00000000_0000_0000_00000000_0000
_0000_00000000_0000_0000_00000000_0000_0000_00000000_0000_0000_00000000_0000_0000_00000000CRLF

Definisce un modello orario con identificatore 5, che consente l'accesso dalle 08:00 alle 13:00 e dalle 14:00 alle 17:00 dal lunedì al venerdì, esclusi gli eventuali giorni festivi (nel caso in cui venga caricato il file CALENDAR.TXT). In questo caso il flag di direzione non è stato usato.

5.9 USERS.TXT

Contiene l'anagrafica degli utenti registrati nel sistema. Ogni record ha una lunghezza fissa di 75 byte (cioè 73 caratteri + CR LF) e contiene i dati di un singolo utente. Opzionalmente si può definire un periodo di validità dell'utente (il cui controllo avviene parallelamente al controllo del periodo di validità della tessera) e associarne l'identificativo ad una particolare tipologia di utente (utile per gestire causali giustificative personalizzate mediante il file AXREASON.TXT, vedi §5.10 a pag. 79) e ad un codice PIN, secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. 70):

IIIIIIIIII_PPPP_NNNNNNNNNNNNNNNNNNN_AAMMGGHHMM_aammgghhmm_E_T_UUUUUUUUUUCR LF

Dove:

IIIIIIIIII	10 byte che rappresentano l'identificatore univoco dell'utente (chiave primaria) a cui viene fatto riferimento nel file CARDS.TXT (vedi §5.4 a pag. 71). Questo dato è richiesto per ogni record e deve essere univoco all'interno del file e diverso da "000…000".
PPPP	4 byte che rappresentano il codice PIN dell'utente. Nota: sono ammesse solo le cifre numeriche '0'..'9', cioè i caratteri ASCII da 30h a 39h. Se questo campo viene impostato a un valore diverso da "0000", e se il parametro **AskPin** all'interno della sezione *[AccessControl]* del file PARAMETERS.TXT (vedi §4.11 a pag. 37) è stato impostato a 1 (default 0), al passaggio di una tessera associata a questo utente il terminale richiederà l'introduzione del PIN, e accetterà la transazione solo se il dato inserito coincide con quello specificato nel record (***Nota***: è anche possibile disabilitare la richiesta del PIN in base alla provenienza della lettura di tessera, vedi parametro **DisableFunctions** al §4.11 a pag. 46). Inoltre, se è abilitata la modalità "solo PIN" (solo sui modelli X2 con tastiera numerica, vedi §5.12 a pag. 81), è possibile per l'utente digitare direttamente il proprio PIN sul terminale per ottenere l'accesso e generare una transazione contenente il codice della tessera, anche se la tessera non è fisicamente presente e quindi in realtà non viene letta. ***Avvertenza 1***: non viene effettuato alcun controllo sull'eventuale presenza nel file USERS.TXT di utenti diversi aventi lo stesso PIN. Ne consegue che gli accessi di tutti gli utenti aventi lo stesso PIN, se effettuati in modalità "solo PIN", daranno luogo a transazioni contenenti sempre e soltanto il codice della tessera associata al primo utente trovato durante la ricerca sequenziale nel file USERS.TXT. ***Avvertenza 2***: il PIN "9999" è riservato e pertanto non può essere utilizzato.
NNNNNNNNNNNNNNNNNNN	20 byte che rappresentano il nome dell'utente, cioè la stringa che viene mostrata nella schermata principale del terminale al posto del codice della tessera in caso di transazione accettata. Questo dato è richiesto per ogni record poiché viene sempre visualizzato, anche se è impostato a "000…000". Poiché la lunghezza del campo è fissa a 20 caratteri, se si desidera inserire un nome più breve è comunque necessario completare il campo inserendo dei caratteri spazio " " (*blank*, 20h). Nota: è possibile utilizzare solo i caratteri visualizzabili delle tabelle Windows-125*x* mostrate al §20 a pag. 177.
AAMMGGHHMM	10 byte che rappresentano la data e ora di inizio del periodo di validità dell'utente. E' equivalente all'analogo campo di CARDS.TXT (vedi §5.4 a pag. 71) e ha lo stesso formato. Nota: l'orario 00:00 corrisponde sempre all'inizio della giornata specificata, mentre 23:59 corrisponde alla fine della giornata (valore massimo utilizzabile). Il controllo sul periodo di

validità dell'utente viene effettuato in parallelo al controllo sul periodo di validità della tessera. Questo dato è facoltativo e può essere semplicemente riempito con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record; in caso contrario, se l'utente transita in un periodo precedente alla data specificata, l'accesso viene rifiutato.

aammgghhmm — 10 byte che rappresentano la data e ora di fine del periodo di validità dell'utente. E' equivalente all'analogo campo di CARDS.TXT (vedi §5.4 a pag. 71) e ha lo stesso formato. Nota: l'orario 00:00 corrisponde sempre all'inizio della giornata specificata, mentre 23:59 corrisponde alla fine della giornata (valore massimo utilizzabile). Il controllo sul periodo di validità dell'utente viene effettuato in parallelo al controllo sul periodo di validità della tessera. Questo dato è facoltativo e può essere semplicemente riempito con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record; in caso contrario, se l'utente transita in un periodo successivo alla data specificata, l'accesso viene rifiutato.

E — 1 byte che rappresenta il flag di abilitazione dell'utente. '1' significa utente abilitato all'accesso, '0' significa utente presente in archivio ma attualmente disabilitato (transito non consentito). Il controllo sull'abilitazione dell'utente viene effettuato in parallelo al controllo sull'abilitazione della tessera.

T — 1 byte che rappresenta la possibilità per l'utente di usufruire di un'estensione dei tempi massimi consentiti per l'apertura e per l'attraversamento di un varco (solo nel caso in cui la gestione avanzata del varco sia attivata, vedi par. §6 a pag. 88). '0' significa utente a cui sono assegnati i tempi standard, definiti dai parametri **TimeOutOpen** e **TimeOutClose** all'interno della sezione *[AccessControl]* del file PARAMETERS.TXT (vedi §4.11 a pag. 38); '1' significa utente a cui sono assegnati i tempi estesi, definiti dagli analoghi parametri **TimeOutOpenExtended** e **TimeOutCloseExtended**.

UUUUUUUUUU — 10 byte che rappresentano la tipologia dell'utente. Questo codice può essere usato nel file AXREASON.TXT (vedi §5.10 a pag. 79) per consentire la selezione di determinate causali giustificative solo ad alcuni utenti. E' un dato facoltativo e può essere semplicemente riempito con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record.

Esempio di record di USERS.TXT:

```
0000000001_0000_Mario Rossi          _1101121830_0000000000_1_0_0000000002CRLF
```

Definisce un utente con identificatore 1, tipologia 2, di nome "Mario Rossi", senza gestione del PIN, abilitato all'accesso (compatibilmente con le autorizzazioni associate alla sua tessera) a partire dalle 08:30 del 12 gennaio 2011 e senza alcuna scadenza, con i tempi standard di apertura e attraversamento del varco.

Nota: solo nel caso in cui X1/X2 venga gestito dal programma Xatlas, esistono anche altri 2 formati possibili per questo file:

- 96 byte (cioè 94 caratteri + CR LF): in questo caso viene aggiunto un ulteriore campo "Identificativo Utente" di 20 caratteri (più il relativo separatore '_' in testa), preso direttamente dall'anagrafica di Xatlas e considerato esclusivamente in fase di stampa di uno scontrino;

- 135 byte (cioè 133 caratteri + CR LF): in questo caso, oltre al campo "Identificativo Utente" descritto al punto precedente, vengono aggiunti 4 ulteriori campi (ciascuno con il relativo separatore '_' in testa), sempre presi dall'anagrafica di Xatlas, che vengono in realtà considerati esclusivamente dai terminali SuperTRAX 7/SuperGLASS 7 con applicazioni Lite e ZTApps in funzione contemporaneamente.

5.10 AXREASON.TXT

Contiene l'elenco delle causali speciali di ingresso / uscita per la rilevazione delle presenze. Ogni record ha una lunghezza fissa di 107 byte (cioè 105 caratteri + CR LF) e associa una singola causale ad un massimo di 4 diversi identificativi di tipologia utente definiti nel file USERS.TXT (vedi §5.9 a pag. 77) secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. 70):

IIIIIIIIII_CCCCCCCCCC_00_DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD_E_UUUUUUUUUU_UUUUUUUUUU_ UUUUUUUUUU_UUUUUUUUUU_FCR LF

Dove:

IIIIIIIIII — 10 byte che rappresentano l'identificatore univoco del record (chiave primaria). Questo dato è richiesto per ogni record e deve essere univoco all'interno del file e diverso da "000…000".

CCCCCCCCCC — 10 byte che rappresentano il codice causale, che deve essere formato da sole cifre puramente numeriche. Se il codice causale che si vuole utilizzare è più corto di 10 cifre è comunque necessario anteporre alla parte significativa del codice tanti caratteri '0' di riempimento fino a raggiungere i 10 byte.

00 — 2 byte che rappresentano il numero di cifre della parte significativa del codice causale, cioè la parte che sarà effettivamente registrata nel file TRANSACTIONS.TXT (vedi §7 a pag. 96). Campo attualmente non gestito: lasciare fisso a '00' (il terminale rimuove automaticamente tutti gli zeri iniziali e mantiene tutte le cifre a partire dalla prima diversa da '0').

DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD — 32 byte che contengono il nome descrittivo della causale. Di questi, i primi 20 caratteri costituiscono la stringa che viene visualizzata nel menu di selezione della causale (a cui si può accedere col tasto ▼) e nella schermata principale a transazione effettuata, mentre i primi 8 vengono mostrati durante la revisione dati di presenza (vedi §10.7 a pag. 118). Questo dato è richiesto per ogni record poiché viene sempre visualizzato, anche se è impostato a "000…000". Poiché la lunghezza del campo è fissa a 32 caratteri, se si desidera inserire un nome più breve è comunque necessario completare il campo inserendo dei caratteri spazio " " (*blank*, 20h). Nota: è possibile utilizzare solo i caratteri visualizzabili delle tabelle Windows-125*x* mostrate al §20 a pag. 177.

E — 1 byte che rappresenta il flag di abilitazione della causale. '1' significa causale abilitata, '0' significa causale presente in archivio ma attualmente disabilitata (causale non visualizzata e quindi non selezionabile nel menu di selezione accessibile col tasto ▼).

UUUUUUUUUU — 10 byte che rappresentano una tipologia di utente abilitata all'utilizzo della causale (definita nel file USERS.TXT, vedi §5.9 a pag. 77). Ad ogni record di AXREASON.TXT si possono associare fino a 4 campi di questo tipo. Se è sufficiente associare una sola tipologia di utenti alla causale, è comunque necessario riempire tutti i campi seguenti con "0000000000" (10 caratteri '0') per mantenere la lunghezza fissa del record. Se la causale deve essere selezionabile da qualunque tipologia di utente, tutti e 4 i campi devono essere riempiti con "0000000000", (10 caratteri '0').

F — 1 byte che rappresenta il flag di "forzatura accesso" associato alla causale. '1' significa che qualora venga selezionata la causale associata a questo flag l'accesso debba essere consentito in ogni caso, indipendentemente dall'esito di tutti gli altri controlli di validità.

Attenzione: il controllo dell'associazione <codice tessera – utente - tipologia utente - causale compatibile> deve comunque essere superato (vedi Nota1 qui sotto), mentre vengono ignorati tutti gli altri (al limite il codice tessera e l'utente possono anche essere disabilitati). '0' significa invece causale con comportamento normale.

Esempio di record di AXREASON.TXT:

```
0000000001_0000000692_00_Pausa pranzo                    _1_0000000002_0000000000_0000000000_
0000000000_0CRLF
```

Definisce una causale con identificatore 1, avente codice 692 e descrizione "Pausa pranzo", abilitata alla visualizzazione nel menu di selezione e utilizzabile dai soli utenti di tipologia 2.

Nota 1: se il controllo accessi è abilitato, la selezione di una causale contenuta in AXREASON.TXT funziona correttamente solo per i codici tessera definiti nel file CARDS.TXT (vedi §5.4 a pag. 71), e associati ad utenti definiti nel file USERS.TXT (vedi §5.9 a pag. 77) e facenti parte di una tipologia abilitata, con un'unica eccezione: che la causale sia associata ad ogni tipologia di utente. In quest'ultimo caso CARDS.TXT e USERS.TXT possono anche non essere presenti (ovviamente deve essere presente almeno il file CARDRNGE.TXT, altrimenti nessun codice tessera sarebbe valido, a meno di usare il flag di "forzatura accesso").

Nota 2: il file AXREASON.TXT ha effetto anche se il controllo accessi non è abilitato. In questo caso tutte le causali ivi specificate (se abilitate con il flag **E**) vengono comunque accettate per ogni codice tessera, anche se sono associate a tipologie di utenti che in realtà non sono definite poiché il file USERS.TXT è assente o non controllato. Inoltre, il file AXREASON.TXT è sempre più prioritario del file REASONS.TXT (vedi §4.4 a pag. 19) usato per definire le causali generiche (sempre valide per tutti gli utenti), indipendentemente dall'abilitazione del controllo accessi. Pertanto, se AXREASON.TXT è presente, REASONS.TXT non viene mai considerato (è come se non ci fosse). In tutti i casi, X1/X2 è in grado di gestire un massimo di 40 causali, pertanto raccomandiamo di non inserire più di 40 record in questi files, poiché i record in eccesso verrebbero comunque ignorati.

5.11 CALENDAR.TXT

Contiene il calendario delle festività personalizzate in cui è possibile abilitare o meno ciascuna fascia oraria definita in TIMEMOD.TXT (vedi §5.8 a pag. 75). Questo file ha un formato leggermente differente dagli altri, in quanto contiene un unico record di 666 byte (cioè 664 caratteri + CR LF) nel quale si possono definire fino a 95 date di festività secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. 70):

GGMMAA_GGMMAA_GGMMAA_...GGMMAACR LF

Dove:

GGMMAA 6 byte che rappresentano la data di ciascuna festività personalizzata in cui è possibile abilitare o meno ciascuna fascia oraria definita in TIMEMOD.TXT (vedi §5.8 a pag. 75). Si devono indicare 2 cifre per il giorno, 2 cifre per il mese e le 2 ultime cifre dell'anno. Nell'unico record di CALENDAR.TXT si possono definire fino a 95 date di festività, coprendo un periodo di almeno 3 anni. Ovviamente non ha alcun senso caricare un file CALENDAR.TXT che non definisca almeno una data significativa. Se non si desidera completare l'inserimento delle 95 date è comunque necessario riempire tutti i campi rimanenti con "000000" (6 caratteri '0') per mantenere la lunghezza fissa del record.

Nota: il file CALENDAR.TXT viene controllato soltanto alla mezzanotte o dopo un riavvio del terminale. Se inizialmente è assente e viene caricato senza riavviare, è come se non ci fosse; analogamente, se viene cancellato senza riavviare, è come fosse ancora presente, e qualsiasi modifica non ha effetto senza un riavvio.

Esempio di CALENDAR.TXT:

251211_010112_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000
000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000
_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_00
0000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_00000
0_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_0
00000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_0000
00_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_000000_
000000CRLF

Definisce il 25 dicembre 2011 ed il 1 gennaio 2012 come giorni festivi.

5.12 LA MODALITA' "SOLO PIN"

Impostando a 1 il parametro **PinOnly** nella sezione *[AccessControl]* del file PARAMETERS.TXT (vedi §4.11 a pag. 36) oppure, analogamente, spuntando la checkbox "**Allow PIN only**" nella pagina "**Access Control**" del web server HTTP, si attiva la modalità "solo PIN" (solo sui modelli X2 con tastiera numerica).

In questo caso l'utente può digitare direttamente il proprio PIN sul terminale per ottenere l'accesso e generare una transazione contenente il codice della tessera, anche se la tessera non è fisicamente presente e quindi in realtà non viene letta. A tale scopo, inoltre, è necessario avere caricato almeno i file CARDS.TXT (vedi §5.4 a pag. 71) e USERS.TXT (vedi §5.9 a pag. 77) dove si è associato il codice tessera ad un utente abilitato. Nota: la modalità "solo PIN" funziona anche se il controllo accessi generico non è stato abilitato (parametro **Enable** nella sezione *[AccessControl]* del file PARAMETERS.TXT, vedi §4.11 a pag. 36), a patto che siano stati caricati entrambi i file sopra menzionati. In questo caso, tuttavia, il nome dell'utente non viene visualizzato (compare solo il codice tessera).

Avvertenza: non viene effettuato alcun controllo sull'eventuale presenza nel file USERS.TXT di utenti diversi aventi lo stesso PIN. Ne consegue che gli accessi di tutti gli utenti aventi lo stesso PIN, se effettuati in modalità "solo PIN", daranno luogo a transazioni contenenti sempre e soltanto il codice della tessera associata al primo utente trovato durante la ricerca sequenziale nel file USERS.TXT.

Si ricordi comunque che è possibile digitare manualmente pure il codice della tessera (anche con il controllo accessi disabilitato), come descritto al §10.2 a pag. 108 (anche in questo caso solo sui modelli X2 con tastiera numerica). L'eventuale attivazione della modalità "solo PIN" è prioritaria rispetto all'attivazione della modalità "digitazione manuale del codice della tessera": se sono attive entrambe, alla pressione di un qualunque tasto si passa direttamente all'introduzione del PIN (non è possibile effettuare manualmente transazioni relative a utenti non associati ad un PIN).

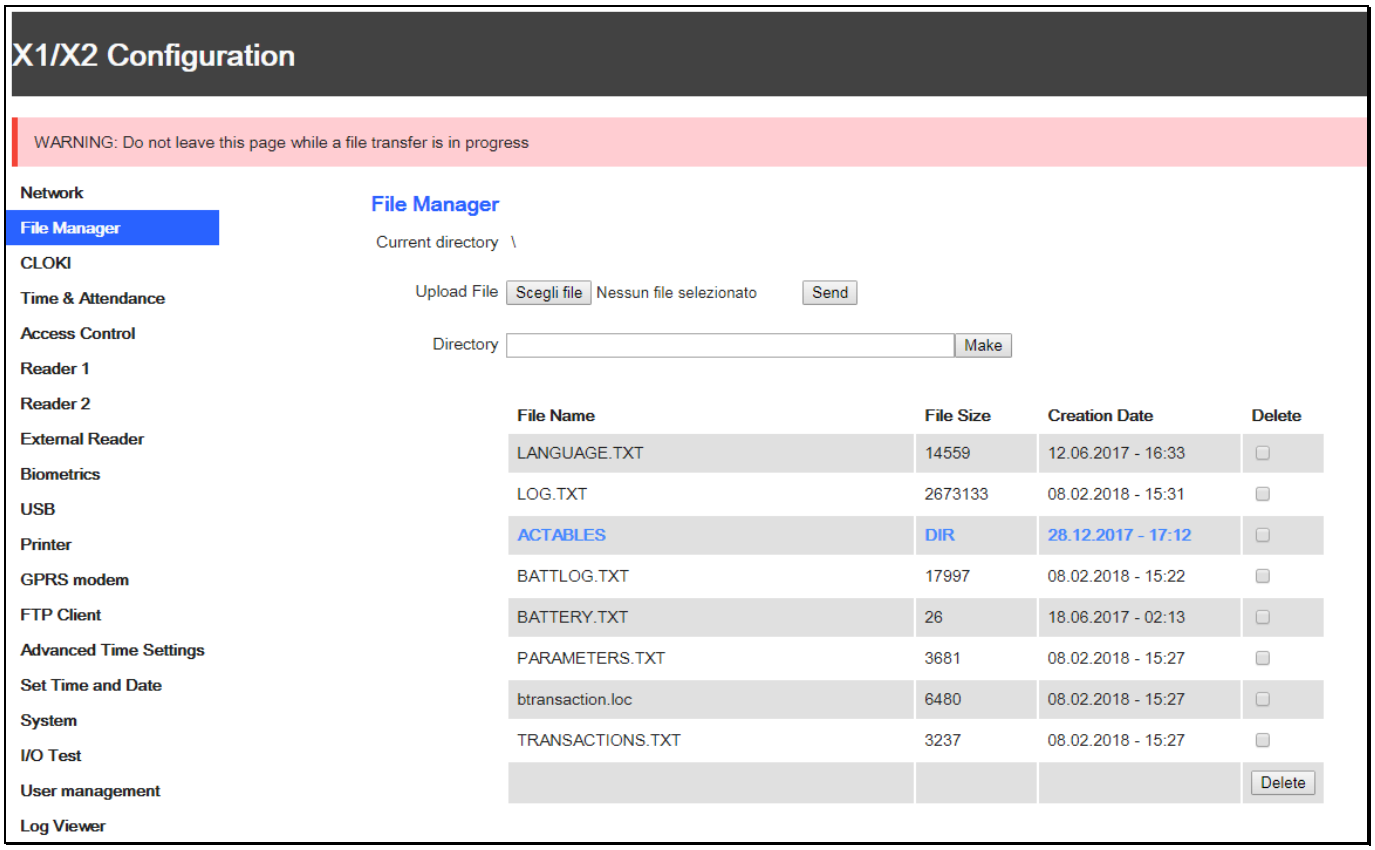
5.13 MESSAGGI DI ERRORE

Messaggio Visualizzato	*Significato*
Tessera non valida	Codice tessera troppo corto *oppure* Codice comune non valido *oppure* Lunghezza del "RAW data" non valida *oppure* File CARDS.TXT e CARDRNGE.TXT assenti *oppure* Codice tessera non presente in CARDS.TXT e non compreso in un intervallo di CARDRNGE.TXT *oppure* Codice tessera compreso in un intervallo di CARDRNGE.TXT disabilitato
Utente disabil.	Codice tessera relativo ad un utente presente in USERS.TXT ma disabilitato
Tessera disabil.	Codice tessera presente in CARDS.TXT ma disabilitata
Tessera scaduta	Codice tessera associato ad un periodo temporale scaduto o non ancora iniziato in CARDS.TXT *oppure* Codice tessera compreso in un intervallo associato ad un periodo temporale scaduto o non ancora iniziato in CARDRNGE.TXT *oppure* Codice tessera relativo ad un utente associato ad un periodo temporale scaduto o non ancora iniziato in USERS.TXT
Non autorizzata	Codice tessera associato tramite AUTHGRP.TXT ad una autorizzazione non presente in AUTH.TXT (se comunque quest'ultimo è stato caricato)
Utente fuori orario	Codice tessera associato tramite TIMEMOD.TXT a fasce orarie scadute o non ancora iniziate (deve valere per tutte le fasce orarie abilitate nel giorno della settimana o festività corrente; se vi siano altre fasce orarie non abilitate nello stesso giorno è irrilevante)
Gruppo aut. assente	Codice tessera associato tramite CARDS.TXT ad un gruppo di autorizzazioni non presente in AUTHGRP.TXT (se comunque quest'ultimo è stato caricato) *oppure* Codice tessera compreso in un intervallo associato tramite CARDRNGE.TXT ad un gruppo di autorizzazioni non presente in AUTHGRP.TXT (se comunque quest'ultimo è stato caricato)
Timemod assente	Codice tessera associato tramite AUTH.TXT ad un modello orario non presente in TIMEMOD.TXT (se comunque quest'ultimo è stato caricato) *oppure* Codice tessera associato tramite AUTH.TXT ad un modello orario presente in TIMEMOD.TXT (in formato "esteso"), ma valido solo per la direzione di transito opposta
Pincode errato	Codice PIN inserito non corrispondente con quello contenuto in USERS.TXT e associato all'utente relativo al codice tessera utilizzato *oppure* E' abilitata la modalità "solo PIN", ma il codice PIN inserito non è presente in USERS.TXT (magari perché USERS.TXT non è stato caricato)

Giorno non valido	Codice tessera associato tramite TIMEMOD.TXT a fasce orarie tutte non abilitate nel giorno della settimana o festività corrente
Edizione non valida	Codice tessera presente in CARDS.TXT, ma il codice edizione contenuto nella tessera non corrisponde a quello specificato per quel codice tessera in CARDS.TXT
Causale non valida	E' stata selezionata una causale definita in AXREASON.TXT ma associata ad una tipologia di utente specifica e non corrispondente a quella dell'utente relativo al codice tessera letto, per uno dei seguenti motivi: Codice tessera compreso in un intervallo abilitato di CARDRNGE.TXT, ma non presente in CARDS.TXT (magari perché CARDS.TXT non è stato caricato) *oppure* Codice tessera presente in CARDS.TXT ma associato ad un utente non presente in USERS.TXT (magari perché USERS.TXT non è stato caricato)
Tab Err: AUTHGRP.TXT	Almeno uno fra AUTH.TXT e TIMEMOD.TXT è stato caricato ma AUTHGRP.TXT è assente
Tab Err: AUTH.TXT	AUTHGRP.TXT è stato caricato ma AUTH.TXT è assente
Tab Err: TIMEMOD.TXT	AUTHGRP.TXT e AUTH.TXT sono stati caricati ma TIMEMOD.TXT è assente

5.14 CLOKI (IN PRECEDENZA DENOMINATO "WEB TABLE EDITOR")

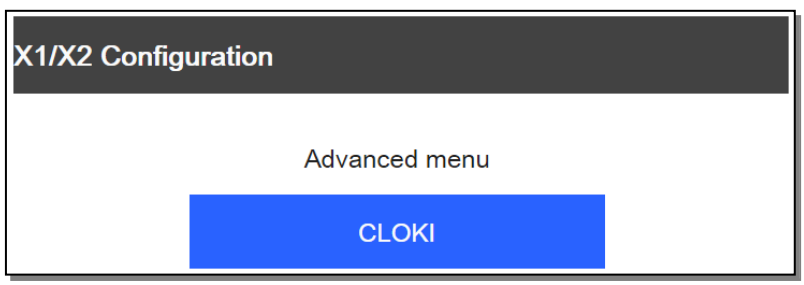
Il web server HTTP di X1/X2 include un web editor per le tabelle di controllo accessi di facile utilizzo chiamato "CLOKI". Si tratta di una estensione del firmware rilasciata separatamente, ma che viene comunque installata in fase di produzione del terminale, e che consiste in una cartella di file chiamata ACTABLES da caricare nella root della micro-SD. Se per qualche motivo la cartella non è stata caricata, o è stata cancellata, o è necessario un aggiornamento ad una versione più recente (previa cancellazione di quella vecchia), dovete solo scaricare dalla nostra area partners il file zippato chiamato ACTABLES_*v.nn*.zip (dove *v* è il numero della versione e *nn* il numero del rilascio), scompattarlo e caricare la cartella ACTABLES risultante su X1/X2 via FTP esattamente com'è, nella root della micro-SD card, con il nome fisso ACTABLES e senza sottocartelle:

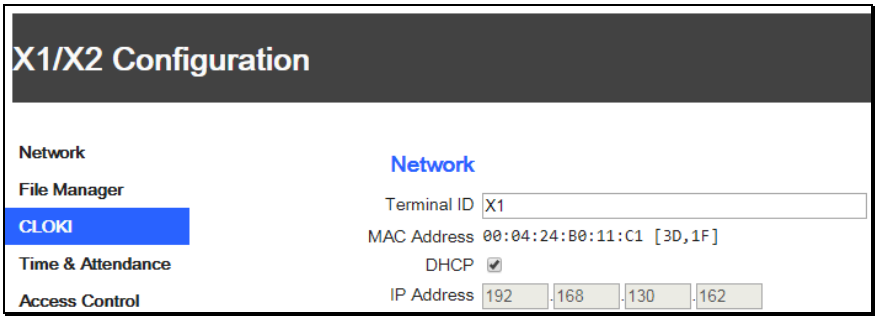


Nota: quanto segue è valido solo per le versioni di CLOKI 1.42 o successive.

Attenzione: nel caso in cui aggiorniate CLOKI ad una versione più recente (cosa che può essere fatta sovrascrivendo la precedente cartella ACTABLES con la nuova), ricordatevi di premere **Ctrl**+**F5** sulla tastiera al primo accesso ad una qualunque pagina del CLOKI aggiornato, in modo da assicurarvi che la cache del browser sia stata ripulita. Questo deve essere fatto per ogni diversa pagina di CLOKI, altrimenti possono verificarsi degli errori durante la modifica dei rispettivi file.

Potete accedere alla sezione CLOKI facendo clic sull'apposito collegamento nella pagina iniziale del web server HTTP, vedi figura a destra: questo è possibile anche nel caso in cui l'utente disponga solo dell'account "manager" (vedi §4 a pag. 14). In alternativa, se siete già entrati nella sezione "Advanced menu" (avendo effettuato l'accesso con l'account "admin"), potete usare il collegamento evidenziato nella figura in basso.

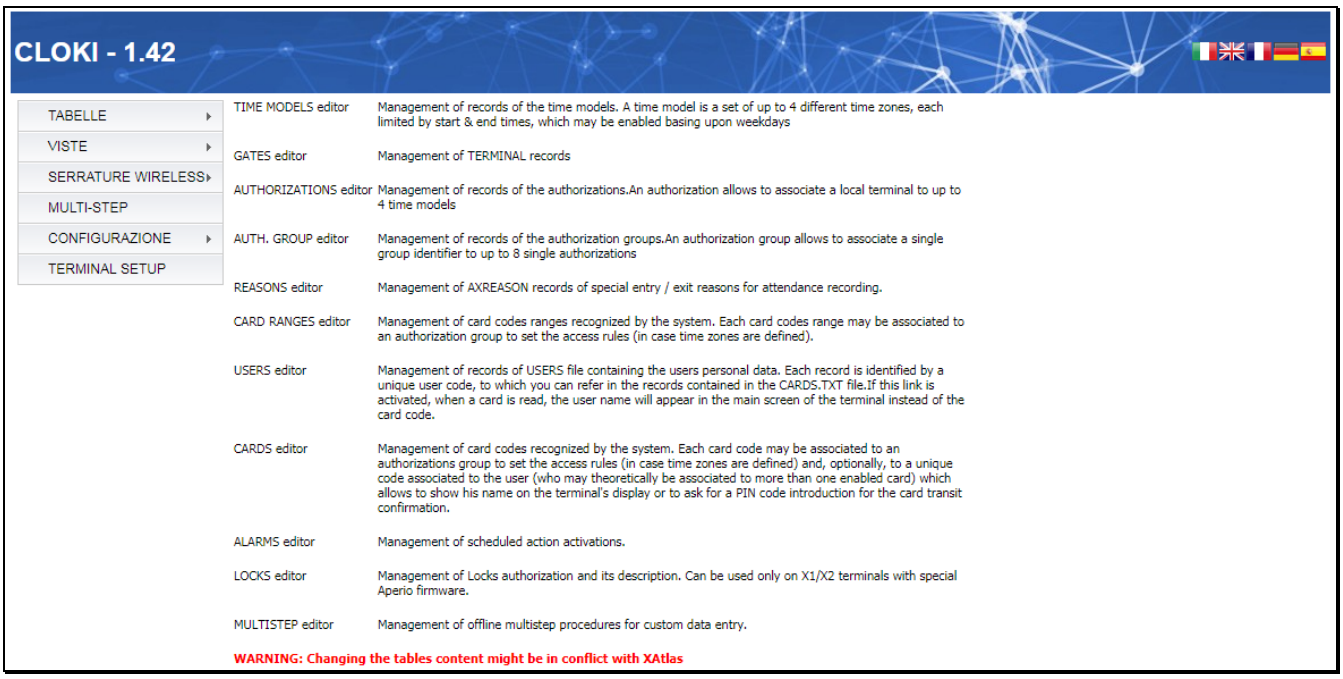




Nota: i collegamenti citati sono comunque sempre presenti, ma generano una pagina di errore se per qualche motivo la cartella ACTABLES non è stata caricata, oppure è stata cancellata).

Le etichette all'interno di ogni pagina dell'editor di tabelle vengono automaticamente tradotte nella lingua correntemente impostata per il browser utilizzato (le opzioni supportate sono: italiano, inglese, francese, tedesco e spagnolo). E' comunque possibile cambiare la lingua per tutta la durata della sessione di comunicazione corrente cliccando la relativa icona fra quelle mostrate in alto a destra.

La pagina principale del CLOKI mostra alcuni collegamenti a sezioni a sezioni sulla sinistra: posizionando il cursore su ciascun collegamento di sezione, verrà visualizzato un elenco di collegamenti di sottosezione accanto ad esso, ciascuno dei quali apre un editor di file diverso: una breve descrizione di ciascun editor di file (disponibile solo in inglese) è mostrata nella pagina principale.



Note:

1) Le sezioni "MULTI-STEP" e "CONFIGURAZIONE" non devono essere utilizzate su X1/X2 poiché sono riservate per l'utilizzo con i terminali X3, SuperGLASS 4 e SuperTRAX/SuperGLASS Light;

2) La sezione "TABELLE" include anche un editor chiamato "ALLARMI" che serve per schedulare degli eventi temporizzati mediante la creazione del file ALARMS.TXT (vedi §4.2 a pag. 18), un editor chiamato "FKEY" che serve per definire i tasti di scelta rapida mediante la creazione del file FKEY.TXT (vedi §4.5 a pag. 20) ed un editor chiamato "VARCHI" che non deve essere utilizzato su X1/X2 poiché è riservato per l'utilizzo con i terminali multi-varco (X3, SuperGLASS 4, Ax-Gate, Ax-Door, XIO e SuperTRAX/SuperGLASS Light);

3) La sezione "VISTE" in realtà non conduce ad un editor di file. Contiene tre collegamenti: il primo ("TRANSAZIONI") mostra tutte le transazioni memorizzate nel terminale in maniera facilmente leggibile (se necessario applicando dei filtri di ricerca per visualizzare solo determinate transazioni), inoltre consente di esportare automaticamente le transazioni di ciascun dipendente singolarmente in formato .PDF o .CSV (***Attenzione***: per abilitare l'esportazione su file .CSV è necessario

consentire esplicitamente i contenuti Flash per la pagina web in questione nel browser utilizzato, cosa che è disabilitata per default); il secondo ("MANUTENZIONE CARTELLINO") consente di modificare il il cartellino mensile di ciascun dipendente (senza comunque perdere le informazioni relative alle transazioni originarie) ed esportarlo, unicamente in formato .PDF; il terzo ("ASSENTI & PRESENTI") mostra quali utenti sono attualmente presenti nell'impianto e quali invece no, basandosi sulle direzioni associate alle ultime transazioni effettuate in un arco di tempo configurabile;

4) La sezione "SERRATURE WIRELESS" è riservata alle versioni di X1/X2 con firmware speciale per la gestione delle serrature wireless Aperio offline, vedi §18 a pag. 165;

5) Il link "TERMINAL SETUP", infine, riporta alla pagina iniziale del web server HTTP.

L'uso di ciascun editor di file è abbastanza intuitivo (naturalmente per il significato dei vari campi potete fare riferimento ai §5.4-5.10), dovete solo seguire queste regole generali:

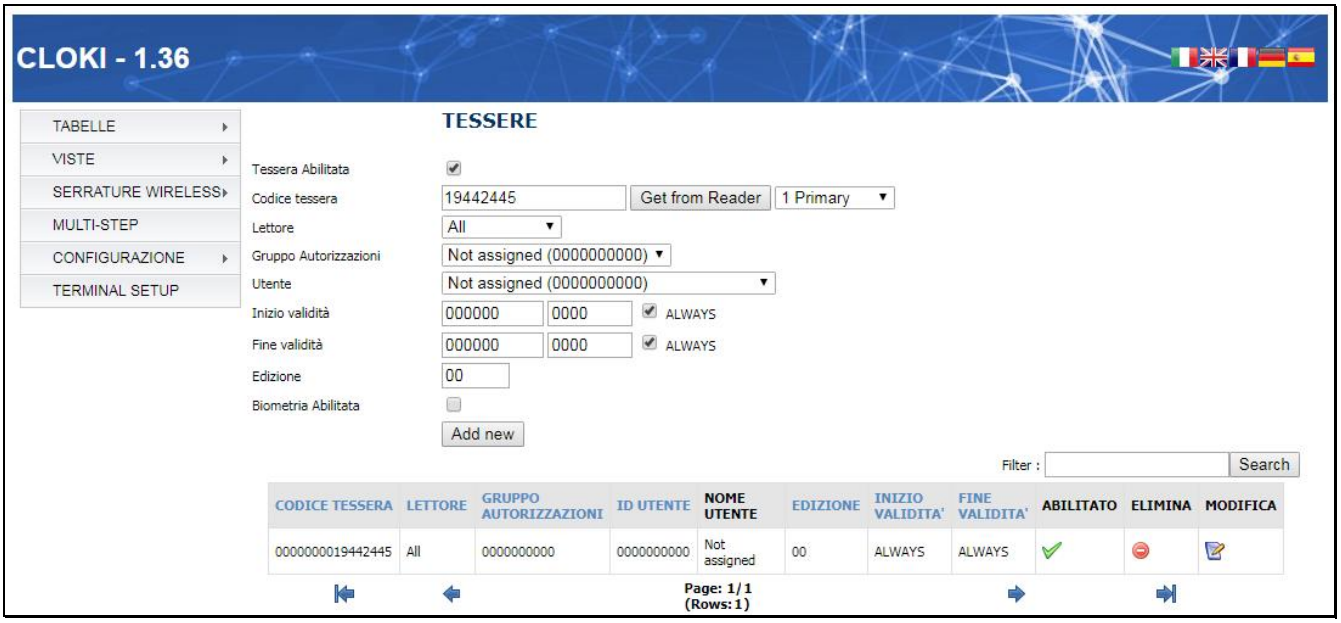
1) Se non dovete usare delle fasce orarie, e se tutti i codici tessera hanno gli stessi diritti di accesso, potete semplicemente usare gli editor "RANGE TESSERE" (per creare CARDRNGE.TXT) oppure "TESSERE" (per creare CARDS.TXT);

2) A causa dei riferimenti incrociati fra i vari file, se dovete definire delle fasce orarie, questo dovrebbe essere fatto come prima cosa, mediante l'editor "MODELLI ORARI" (per creare TIMEMOD.TXT e, opzionalmente, CALENDAR.TXT mediante la sezione "FESTIVITA'"). A questo punto dovrete poi definire le autorizzazioni mediante l'editor "AUTORIZZAZIONI" (per creare AUTH.TXT) e quindi i gruppi di autorizzazioni mediante l'editor "GRUPPO AUTORIZZAZIONI" (per creare AUTHGRP.TXT). Solo a questo punto, potrete usare gli editor "RANGE TESSERE" oppure "TESSERE". L'editor "UTENTI" (per creare USERS.TXT) è opzionale, ma se avete intenzione di usarlo questo va fatto prima di usare l'editor "TESSERE". Per rendere più facile il seguire la corretta sequenza delle operazioni, tutti i link agli editor sulla sinistra sono stati disposti dall'alto al basso secondo l'ordine sopra descritto (***Nota***: l'editor "VARCHI" va utilizzato solo sui terminali Ax-Door / Ax-Gate / X3 / SuperGLASS 4 e SuperTRAX/SuperGLASS Light, poiché i terminali X1/X2 gestiscono sempre un singolo varco il cui identificatore è '1').

Limitazioni: per semplificare il CLOKI, sono applicate le seguenti limitazioni:

- L'editor "MODELLI ORARI" consente di definire solo 4 fasce orarie per ciascun modello orario, invece di 24 come sarebbe teoricamente possibile mediante il file TIMEMOD.TXT
- L'editor "AUTORIZZAZIONI" consente di associare solo 4 modelli orari a ciascuna autorizzazione, invece di 8 come sarebbe teoricamente possibile mediante il file AUTH.TXT
- L'editor "GRUPPO AUTORIZZAZIONI" consente di associare solo 8 autorizzazioni a ciascun gruppo di autorizzazioni, invece di 32 come sarebbe teoricamente possibile mediante il file AUTHGRP.TXT
- L'editor "UTENTI" non consente di definire la tipologia dell'utente, come sarebbe teoricamente possibile mediante il file USERS.TXT; per la stessa ragione, l'editor "CAUSALI" crea in realtà un file AXREASON.TXT i cui record si riferiscono a causali che possono essere selezionate da qualunque utente.

Fra le varie opzioni, l'editor del file CARDS.TXT include un utile pulsante "Get from Reader" che consente di inserire un nuovo codice tessera direttamente da un lettore collegato a X1/X2. Dovete prima scegliere il numero del lettore dal menu a tendina sulla destra: "**1 Primary**", "**2 Secondary**" o "**3 External**" (quest'ultima voce può anche essere usata per le letture effettuate su un lettore aggiuntivo collegato ad una scheda di espansione 914 NeoMAX opzionale, vedi §3.6 a pag. 11). Una volta fatto questo, premete il pulsante ed effettuate la lettura entro 5 secondi: il campo "Card code" verrà riempito con il codice personale estratto in base all'attuale configurazione della relativa sezione *[Reader1]*, *[Reader2]* o *[ExtReader]* del file PARAMETERS.TXT.



Attenzione: in tutti gli editor delle varie tabelle, nella compilazione dei campi di testo evitate di inserire qualunque carattere ASCII superiore al 7Eh, ovvero chr(126)='~' (vedi anche le tabelle di codifica dei caratteri al §20 a pag. 177), incluso ogni tipo di lettera accentata, poiché questo potrebbe generare dei problemi di codifica e conseguente errato formato dei record creati.

6. GESTIONE AVANZATA DI UN VARCO

La logica di gestione del varco viene implementata autonomamente dal terminale in seguito alla lettura di una tessera, all'attivazione di uno degli ingressi digitali disponibili (vedi §6.1 e §6.2) o alla ricezione di un opportuno comando inviato dal server (vedi §6.5 a pag. 93), in base ai criteri definiti da una serie di parametri.

Come si è visto ai §3.3 e §3.4 a pag. 9, il terminale X1/X2 preso singolarmente dispone di 1 sola uscita relé e di 2 ingressi digitali IN1 e IN2, ma collegandovi fino a 2 schede di espansione 914 NeoMAX opzionali è possibile aggiungere fino a 4 uscite relé e 4 ingressi digitali, per un totale di 5 uscite relé (numerate da 1 a 5) e 6 ingressi digitali (numerati da 1 a 6).

Nota importante: la gestione di base di un varco, limitatamente all'attivazione dei relé per lo sblocco in base alla direzione di transito (parametro **RelayActivation**, vedi §4.11 a pag. 36, e parametri **EntryRelay** e **ExitRelay**, vedi §6.4 a pag. 92), è in ogni caso sempre attiva, ma è anche possibile attivare una gestione avanzata che prevede l'utilizzo congiunto di determinati ingressi digitali e/o uscite relé allo scopo di implementare funzioni aggiuntive, come ad esempio lo sblocco manuale del varco da pulsante, lo sblocco o il blocco permanente in condizioni particolari, il controllo dello stato del varco (aperto / chiuso), l'attivazione di allarmi, ecc.

La funzionalità di gestione avanzata di un varco di controllo accessi si attiva impostando a 1 il parametro **GateEnabled** nella sezione *[AccessControl]* del file PARAMETERS.TXT (vedi §4.11 a pag. 37) oppure, analogamente, spuntando la checkbox "**Gate Enabled**" nella pagina "**Access Control**" del web server HTTP. **Note**: su vecchie versioni di firmware (precedenti alla "a12_build802") questa funzione può essere sbloccata solo introducendo un'apposita chiave di attivazione FW (vedi §4.12 a pag. 65), a meno che X1/X2 non sia gestito dal programma Xatl@s (in tal caso la chiave di attivazione viene caricata in maniera automatica).

I parametri per la gestione avanzata del varco si trovano anch'essi nella sezione *[AccessControl]* del file PARAMETERS.TXT (vedi §4.11 a pag. 37) e si possono suddividere nei seguenti gruppi: 1) parametri per la definizione del tipo di varco; 2) parametri per la definizione dei tempi massimi consentiti per il passaggio; 3) parametri per l'assegnazione degli ingressi digitali; 4) parametri per l'assegnazione delle uscite relé. Vediamo nel dettaglio il funzionamento di ciascun parametro.

Nel seguito viene descritto per ciascun gruppo di parametri il relativo funzionamento in caso di modalità offline, o di server non in linea. Per il funzionamento in modalità online, si veda il §6.5 a pag. 93.

6.1 TIPO DI VARCO

- **GateType**

Definisce il tipo di varco da gestire:

0 → varco non controllato (default). Se la gestione avanzata del varco è attivata (par. **GateEnabled**=1) ma il varco non è controllato, l'unica differenza rispetto al caso di gestione avanzata del varco non attivata è che il messaggio "*Keep Alive*" inviato periodicamente all'host (vedi §6.5 a pag. 93) contiene sempre il *server tag* "**&gateStatus=H**" (cioè "stato normale")

1 → porta battente: in questo caso, se non diversamente specificato impostando il parametro **GateSensor1** ad un valore diverso da '1' e assegnando il valore '1' a uno degli altri parametri elencati al §6.3 a pag. 90, l'input IN1 viene automaticamente assegnato allo stato della porta

2 → tornello: in questo caso, se non diversamente specificato impostando il parametro **GateSensor1** ad un valore diverso da '1' e assegnando il valore '1' a uno degli altri parametri elencati al §6.3 a pag. 90, l'input IN1 viene automaticamente assegnato allo stato del tornello

3 → doppia porta o bussola di sicurezza: in questo caso, se non diversamente specificato impostando il parametro **GateSensor1** ad un valore diverso da '1' ed il parametro **GateSensor2** ad un valore diverso da '2', e assegnando i valori '1' e '2' ad altri due parametri fra quelli elencati al §6.3 a pag. 90, l'input IN1 viene automaticamente assegnato allo stato della prima porta, e l'input IN2 allo stato della seconda porta

4 → centrale allarme: valore riservato alla gestione di X1/X2 da parte del programma Xatl@s

Come si può vedere, in tutti i casi di varco controllato, per default l'input IN1 (già disponibile su X1/X2) viene utilizzato per segnalare lo stato del varco (aperto / chiuso), ma è comunque possibile assegnarlo ad un controllo di tipo diverso mediante i parametri elencati al §6.3 a pag. 90. Solo nel caso del varco di tipo doppia porta o bussola di sicurezza, inoltre, per default anche l'IN2 (anch'esso già disponibile su X1/X2) viene usato per lo stesso scopo ma relativamente alla seconda porta, e comunque anch'esso potrà essere riassegnato.

E' anche possibile definire lo stato a riposo degli input utilizzati per segnalare lo stato del varco (cioè per default IN1 e IN2 - quest'ultimo solo nel caso di doppia porta o bussola di sicurezza - ma anche qualunque altro input assegnato a tale scopo se diversamente specificato) mediante i seguenti parametri:

- **GateState1**:

0 → Input aperto (non attivo) con varco chiuso

1 → Input cortocircuitato (attivo) con varco chiuso (default)

- **GateState2** (da usare solo nel caso di doppia porta o bussola di sicurezza):

0 → Input aperto (non attivo) con varco chiuso

1 → Input cortocircuitato (attivo) con varco chiuso (default)

Gli ingressi utilizzati per lo stato del varco (aperto / chiuso) vengono sempre controllati in seguito ad una transazione valida (vedi successivo §6.2 a pag. 89) o ad uno sblocco comandato (vedi §6.3 a pag. 90 e §6.5 a pag. 93), ma anche in assenza di questi eventi per segnalare un'eventuale situazione di forzatura del varco, quando cioè avviene un cambiamento di stato a varco teoricamente chiuso. In questo caso X1/X2 mostra il messaggio "**Effrazione**" e registra nel file TRANSACTIONS.TXT un record relativo all'emissione di un evento, con il campo **EVENTO**="11" (vedi §7.1 a pag. 100). Il messaggio permane fino alla richiusura del varco: a quel punto verrà registrato in TRANSACTIONS.TXT un record relativo al rientro dell'evento "11".

6.2 TEMPI MASSIMI CONSENTITI PER IL PASSAGGIO

In seguito ad una transazione valida, X1/X2 sblocca il varco (si veda il §6.4 per definire quale relé debba essere attivato a tale scopo), mostra il messaggio "**Entrata:** *<codice o nome utente>***"** oppure "**Uscita:** *<codice o nome utente>*" (a seconda della direzione impostata per il lettore utilizzato).

X1/X2 non registra comunque nulla fino al completamento dell'attraversamento del varco o, se questo non avviene, la scadenza di un timeout. A quel punto il record relativo alla transazione ed al suo esito viene registrato in TRANSACTIONS.TXT: se il transito è stato completato nel tempo previsto, tale record ha i campi **CONTROLLI**="00" e **ESITO**="0", altrimenti il campo **ESITO** assume il valore "1".

Vediamo ora quali sono i timeout relativi alla gestione avanzata del varco ed i parametri che li definiscono:

- **TimeOutOpen**

  Definisce il tempo massimo (in decimi di secondo) entro il quale deve essere fisicamente aperto il varco per iniziare l'attraversamento dopo lo sblocco in seguito ad una transazione valida. Default: 50 (5 secondi). In caso di scadenza del timeout il terminale mostra il messaggio "**Nessun Transito**" e genera un record con i campi **CONTROLLI**="00" e **ESITO**="1" (vedi §7 a pag. 96).

- **TimeOutOpenExtended**

  Come il precedente parametro **TimeOutOpen**, ma valido solo per gli utenti a cui è consentito un tempo più lungo per effettuare l'accesso (vedi flag 'T' nei record del file USERS.TXT, §5.9 a pag. 77). Default: 100 (10 secondi).

- **TimeOutClose**

  Definisce il tempo massimo (in decimi di secondo) entro il quale deve essere fisicamente richiuso il varco a partire dal momento in cui viene aperto in seguito ad una transazione valida. Default: 50 (5 secondi). In caso di scadenza del timeout il terminale mostra il messaggio "**Varco non richiuso**" e genera un record con i campi **CONTROLLI**="00" e **ESITO**="1" (vedi §7 a pag. 96). Inoltre nello stesso file viene registrato anche un record relativo all'emissione di un evento, con il campo **EVENTO**="12" (vedi §7.1 a pag. 100). Il messaggio permane fino alla richiusura del varco: a quel punto verrà registrato un altro record relativo al rientro dell'evento "12".

- **TimeOutCloseExtended**

Come il precedente parametro **TimeOutClose**, ma valido solo per gli utenti a cui è consentito un tempo più lungo per effettuare l'accesso (vedi flag 'T' nei record del file USERS.TXT, §5.9 a pag. 77). Default: 100 (10 secondi).

6.3 ASSEGNAZIONE DEGLI INGRESSI DIGITALI

Come visto al §6.1 a pag. 88, in tutti i casi di varco controllato, per default l'input IN1 (già disponibile su X1/X2) viene utilizzato per segnalare lo stato del varco (aperto / chiuso), ma è comunque possibile assegnarlo ad un controllo di tipo diverso mediante uno dei restanti 10 parametri elencati in questo paragrafo. Solo nel caso del varco di tipo doppia porta o bussola di sicurezza, inoltre, per default anche l'IN2 (anch'esso già disponibile su X1/X2) viene usato per lo stesso scopo ma relativamente alla seconda porta, e comunque anch'esso potrà essere riassegnato.

Su un terminale X1/X2 preso singolarmente, il quale dispone di 2 soli ingressi digitali IN1 e IN2, soltanto 2 degli 11 parametri seguenti possono essere usati per l'assegnazione dello stato del varco (necessario) e dell'unico altro ingresso rimanente (e gli unici valori consentiti in questo caso saranno "1" e "2"). Nel caso particolare di varco di tipo doppia porta o bussola di sicurezza, invece, il solo X1/X2 non sarebbe comunque in grado di gestire l'apertura di 2 porte diverse con 1 sola uscita relé.

Usando una scheda di espansione 914 NeoMAX opzionale (vedi §3.4 a pag. 10) con indirizzo RS485 '1' è però possibile aggiungere altri 2 ingressi digitali, per un totale di 4 ingressi digitali. Con questa configurazione, fino a 4 degli 11 parametri seguenti possono essere usati per l'assegnazione dello stato del varco (necessario) e dei 3 ingressi rimanenti (2 nel caso di varco di tipo doppia porta o bussola di sicurezza), infatti anche i valori "3" e "4", relativi agli ingressi digitali aggiuntivi, sono così consentiti.

Infine, usando una ulteriore scheda di espansione 914 NeoMAX con indirizzo RS485 '2', è possibile aggiungere 2 ulteriori ingressi digitali, per un totale di 6 ingressi digitali. Con questa configurazione, fino a 6 degli 11 parametri seguenti possono essere usati per l'assegnazione dello stato del varco (necessario) e dei 5 ingressi rimanenti (4 nel caso di varco di tipo doppia porta o bussola di sicurezza), infatti l'intervallo dei valori consentiti si estende in questo modo fino a "1"..."6".

In generale, per tutti i parametri qui elencati, i valori ammessi sono i seguenti:

0 → non gestito (default per tutti eccetto **GateSensor1**)

1 → input IN1, già disponibile su X1/X2 (default per il parametro **GateSensor1**, se non diversamente specificato impostando il parametro **GateSensor1** ad un valore diverso da '1' e assegnando il valore '1' a uno degli altri parametri seguenti)

2 → input IN2, già disponibile su X1/X2 (default per il parametro **GateSensor2** in caso di impostazione di varco di tipo doppia porta / bussola di sicurezza, se non diversamente specificato impostando il parametro **GateSensor2** ad un valore diverso da '2' e assegnando il valore '2' a uno degli altri parametri seguenti)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1'

5, 6 → input disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

- **GateSensor1**

Input usato per controllare lo stato del varco (aperto/chiuso). Default: 1. Lo stato a riposo dell'input che è stato assegnato a questo scopo è definito dal parametro **GateState1** (vedi §6.1 a pag. 88).

- **GateSensor2**

Input usato per controllare lo stato della seconda porta (aperta/chiusa). Default: 0. Default in caso di impostazione varco di tipo doppia porta o bussola di sicurezza: 2. Lo stato a riposo dell'input che è stato assegnato a questo scopo è definito dal parametro **GateState2** (vedi §6.1 a pag. 88).

- **ManualUnlockIN**

Input usato per gestire un pulsante di sblocco manuale del varco per una singola entrata (sarà attivato il relé definito dal parametro **EntryRelay**, vedi §6.4 a pag. 92, per il tempo definito dal parametro **RelayActivation**, vedi §4.11 a pag. 36). Il terminale mostra il messaggio "**Entrata**" senza alcun codice, e a transito completato o timeout scaduto registra nel file TRANSACTIONS.TXT una transazione in cui il campo **CODICE_UTENTE** viene riempito con soli zeri "000..000", **SOURCE**="8", **DIREZIONE**="1", **CONTROLLI**="00" e **ESITO**="0" se il transito è stato completato correttamente oppure "1" se è scaduto il timeout (vedi §7 a pag. 96).

- **ManualUnlockOUT**

  Input usato per gestire un pulsante di sblocco manuale varco per una singola uscita (sarà attivato il relé definito dal parametro **ExitRelay**, vedi §6.4 a pag. 92, per il tempo definito dal parametro **RelayActivation**, vedi §4.11 a pag. 36). Il terminale mostra il messaggio "**Uscita**" senza alcun codice, e a transito completato o timeout scaduto registra nel file TRANSACTIONS.TXT una transazione in cui il campo **CODICE_UTENTE** viene riempito con soli zeri "000..000", **SOURCE**="8", **DIREZIONE**="0", **CONTROLLI**="00" e **ESITO**="0" se il transito è stato completato correttamente oppure "1" se è scaduto il timeout (vedi §7 a pag. 96).

- **Emergency**

  Input usato per gestire un pulsante di sblocco manuale continuo del varco e segnalazione di emergenza (vengono attivati entrambi i relé definiti dai parametri **EntryRelay** e **ExitRelay**, ammesso che siano diversi, ed anche il relé eventualmente definito dall'omologo parametro **EmergencyRelay**, vedi §6.4 a pag. 92). Il terminale mostra il messaggio "**Emergenza**" e registra nel file TRANSACTIONS.TXT un record relativo all'emissione di un evento, con il campo **EVENTO**="07" (vedi §7.1 a pag. 100). Lo sblocco, e quindi l'attivazione dei relé, permane per tutto il tempo di attivazione dell'input, cioè fintanto che viene premuto il pulsante: a quel punto verrà registrato un altro record relativo al rientro dell'evento "07".

- **GateLocked**

  Input usato per gestire un pulsante di blocco manuale continuo del varco. Il terminale mostra il messaggio "**Non disponibile**" e registra nel file TRANSACTIONS.TXT un record relativo all'emissione di un evento, con il campo **EVENTO**="03" (vedi §7.1 a pag. 100), inoltre attiva il relé eventualmente definito dall'omologo parametro **GateLockedRelay**, vedi §6.4 a pag. 92). Il blocco permane per tutto il tempo di attivazione dell'input, cioè fintanto che viene premuto il pulsante: a quel punto verrà registrato un altro record relativo al rientro dell'evento "03".

- **InterLocked**

  Input usato per bloccare il terminale finché il varco è impegnato poiché è in corso un transito in direzione opposta (il terminale mostra il messaggio "**Varco occupato**"). Utile nel caso in cui vengano usati 2 terminali diversi sui lati opposti di uno stesso varco. Questo input deve essere collegato all'altro terminale sull'uscita relé definita dall'omologo parametro **GateBusy** descritto al §6.4 a pag. 92. Il blocco permane per tutto il tempo di attivazione dell'input.

- **ExternalNoTransit**

  Ha effetto solo se è stato selezionato un varco del tipo tornello (parametro **GateType**=2, vedi §6.1 a pag. 88): definisce l'input usato per ricevere una segnalazione di transito non avvenuto da una logica esterna, normalmente usata nei tornelli. Il terminale mostra immediatamente il messaggio "**Nessun Transito**" e registra nel file TRANSACTIONS.TXT un record relativo alla transazione non completata con i campi **CONTROLLI**="00" e **ESITO**="1" (vedi §7 a pag. 96).

- **TurnstileAlert**

  Ha effetto solo se è stato selezionato un varco del tipo tornello (parametro **GateType**=2, vedi §6.1 a pag. 88): definisce l'input usato per ricevere una segnalazione di effrazione da una logica esterna, normalmente usata nei tornelli. Il terminale mostra immediatamente il messaggio "**Allarme**" e registra nel file TRANSACTIONS.TXT un record relativo all'emissione di un evento, con il campo **EVENTO**="11" (vedi §7.1 a pag. 100). Il messaggio permane fino alla disattivazione dell'input: a quel punto verrà registrato un altro record relativo al rientro dell'evento "11".

I restanti 2 parametri vanno usati solo uno in alternativa all'altro e hanno effetto solo se è stato selezionato un varco del tipo doppia porta o bussola di sicurezza (parametro **GateType**=3, vedi §6.1 a pag. 88):

- **SecurityBoothAuth**

  Definisce l'input usato per ricevere, da parte di una logica esterna normalmente usata nelle bussole, la segnalazione relativa al fatto che è possibile aprire la seconda porta. Questo parametro va usato solo se l'uscita della logica esterna è normalmente bassa (input aperto, cioè non attivo, a riposo). In caso contrario, occorre usare, in alternativa, il seguente parametro **SecurityBoothAuthDeny**.

- **SecurityBoothAuthDeny**

  Definisce l'input usato per ricevere, da parte di una logica esterna normalmente usata nelle bussole, la segnalazione relativa al fatto che non è ancora possibile aprire la seconda porta. Questo parametro va usato solo se l'uscita della logica esterna è normalmente alta (input cortocircuitato, cioè attivo, a riposo). In caso contrario, occorre usare, in alternativa, il precedente parametro **SecurityBoothAuth**.

6.4 ASSEGNAZIONE DELLE USCITE RELE'

In tutti i casi di gestione di un varco, le uniche uscite relé che devono necessariamente essere assegnate sono quelle usate per l'apertura del varco, definite dai parametri

- **EntryRelay**
- **ExitRelay**

rispettivamente per le transazioni in entrata e per quelle in uscita. I valori ammessi sono i seguenti:

1 → relé interno già disponibile su X1/X2 (default)

2, 3 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1'

4, 5 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

Solo per questi 2 parametri è anche possibile usare lo stesso valore (infatti il valore di default è "1" per entrambi), con l'eccezione del caso di varco di tipo doppia porta o bussola di sicurezza, in cui ciascuno controlla l'apertura di una porta diversa.

Ne consegue che su un terminale X1/X2 preso singolarmente, il quale dispone di 1 sola uscita relé, **EntryRelay** e **ExitRelay** devono necessariamente essere impostati entrambi a "1", e nessuno dei 6 parametri seguenti può essere usato per assegnare ulteriori uscite relé. Nel caso particolare di varco di tipo doppia porta o bussola di sicurezza, invece, il solo X1/X2 non è comunque in grado di gestire l'apertura di 2 porte diverse con 1 sola uscita relé.

Usando una scheda di espansione 914 NeoMAX opzionale (vedi §3.3 a pag. 9) con indirizzo RS485 '1' è però possibile aggiungere altre 2 uscite relé, per un totale di 3 uscite relé. Anche i valori "2" e "3", relativi alle uscite relé aggiuntive, possono infatti essere assegnati ai parametri. Con questa configurazione, se **EntryRelay** e **ExitRelay** vengono impostati allo stesso valore, fino a 2 dei 6 parametri seguenti possono essere usati per l'assegnazione delle 2 uscite relé non già utilizzate (e dovranno avere valori diversi fra loro e diversi da quello di **EntryRelay** e **ExitRelay**). Se invece **EntryRelay** e **ExitRelay** vengono impostati a valori diversi, solo 1 dei 6 parametri seguenti può essere usato per l'assegnazione dell'unica uscita relé non già utilizzata (e dovrà avere un valore diverso da quelli di **EntryRelay** e **ExitRelay**).

Infine, usando una ulteriore scheda di espansione 914 NeoMAX con indirizzo RS485 '2', è possibile aggiungere 2 ulteriori uscite relé, per un totale di 5 uscite relé. L'intervallo dei valori consentiti per i parametri relé si estende in questo modo fino a "1"..."5". Con questa configurazione, se **EntryRelay** e **ExitRelay** vengono impostati allo stesso valore, fino a 4 dei 6 parametri seguenti possono essere usati per l'assegnazione delle 4 uscite relé non già utilizzate (e dovranno avere valori diversi fra loro e diversi da quello di **EntryRelay** e **ExitRelay**). Se invece **EntryRelay** e **ExitRelay** vengono impostati a valori diversi, solo 3 dei 6 parametri seguenti possono essere usati per l'assegnazione delle 3 uscite relé non già utilizzate (e dovranno avere valori diversi da quelli di **EntryRelay** e **ExitRelay**).

In generale, per tutti i parametri qui elencati, i valori ammessi sono i seguenti:

0 → non gestito (default)

1 → relé interno già disponibile su X1/X2

2, 3 → relé esterni, disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '1'

4, 5 → relé esterni disponibili solo su scheda 914 NeoMAX opzionale con indirizzo RS485 '2'

- **DeniedRelay**

  Relé da attivare per segnalare (ad esempio tramite una luce o un segnalatore acustico) un tentativo di transazione non valido. L'attivazione di questo relé persiste per il tempo specificato dal parametro **DeniedRelayTimeout** (vedi §4.7 a pag. 36).

- **EmergencyRelay**

  Relé da attivare per segnalare (ad esempio tramite una luce o un segnalatore acustico) la situazione di emergenza generata dall'attivazione manuale dell'input associato all'omologo parametro **Emergency** (vedi §6.3 a pag. 90). L'attivazione di questo relé permane per tutto il tempo di attivazione dell'input.

- **GateLockedRelay**

Relé da attivare per segnalare (ad esempio tramite una luce o un segnalatore acustico) la situazione di blocco del varco generata dall'attivazione manuale dell'input associato all'omologo parametro **GateLocked** (vedi §6.3 a pag. 90). L'attivazione di questo relé permane per tutto il tempo di attivazione dell'input.

- **GateAlert**

Relé da attivare per segnalare, ad esempio tramite una luce o un segnalatore acustico, la situazione di allarme generata da un cambiamento di stato di un input relativo ad un sensore di stato varco quando il varco è chiuso (varco forzato), o dall'attivazione dell'input associato all'omologo parametro **TurnstileAlert** (effrazione, solo in caso di varco di tipo tornello, vedi §6.3 a pag. 88). L'attivazione di questo relé permane per tutto il tempo di attivazione dell'input.

- **GateTransitOk**

Relé da attivare per segnalare (ad esempio tramite una luce o un segnalatore acustico) la situazione di varco sbloccato in seguito ad una transazione valida o all'attivazione degli input associati ai parametri **ManualUnlockIN** e **ManualUnlockOUT** (sblocco manuale da pulsante per singola entrata o uscita, vedi §6.3 a pag. 88). L'attivazione di questo relé permane dallo sblocco del varco fino a transito ultimato (timeout scaduto senza apertura varco, cioè transito non effettuato, oppure varco richiuso).

- **GateBusy**

Relé da attivare per segnalare che il varco è impegnato. Utile nel caso in cui vengano usati 2 terminali diversi sui lati opposti di uno stesso varco. Questa uscita relé deve essere collegata all'altro terminale sull'input definito dall'omologo parametro **InterLocked** (vedi §6.3 a pag. 88) per segnalargli che non è possibile effettuare transiti. L'attivazione del relé permane dallo sblocco del varco fino a transito ultimato (timeout scaduto senza apertura varco, cioè transito non effettuato, oppure varco richiuso).

6.5 GESTIONE ONLINE DEL VARCO

Quanto descritto nel seguito è valido solo nel caso in cui la gestione dei messaggi con protocollo HTTP sia stata attivata (vedi §12 a pag. 137; X1/X2 può essere gestito in modalità online anche mediante il programma Xatl@s, ma in questo caso utilizza un protocollo diverso). Quando funziona in modalità online e la gestione avanzata del varco è attiva, X1/X2 comunica periodicamente al server quale sia l'attuale stato del varco, aggiungendo la stringa "**&gateStatus=*X***" in coda ai messaggi "*keep alive*" descritti al §12.3 a pag. 140, dove '*X*' è una lettera maiuscola che può assumere i seguenti valori e significati:

C Varco disabilitato (bloccato, corrispondente all'evento "03")
F Varco libero (sbloccato indefinitamente, esclusivamente su invio comando online **gateCmd=free**, vedi nel seguito)
G Varco in emergenza (corrispondente all'evento "07")
H Stato normale (chiuso e a riposo)
I Varco occupato (in seguito a transazione valida o sblocco singolo, fino a transito completato o timeout scaduto)
K Varco forzato (aperto a riposo, corrispondente all'evento "11")
L Varco non richiuso dopo apertura comandata (corrispondente all'evento "12")
N Notifica stato varco disabilitata (esclusivamente su invio comando online **gateCmd=no_ctrl**, vedi nel seguito)

Se la gestione è attiva ma il varco è impostato come "non controllato" (parametro **GateType**=0, vedi §6.1 a pag. 88), l'unico valore utilizzato sarà sempre 'H'.

Per quanto riguarda i messaggi mostrati sul display in caso di evento varco non cambia nulla rispetto al caso offline. Tuttavia, in modalità online non vengono mai creati record relativi ad un evento all'interno del file TRANSACTIONS.TXT. Al contrario, ogni volta che cambia lo stato del varco, le segnalazioni di "emissione" e di "rientro" (cioè il ritorno allo stato normale) del relativo evento vengono sempre affidate a messaggi "*keep alive*" generati subito dopo ciascun cambio di stato (quindi senza attendere lo scadere dell'intervallo che normalmente intercorre fra un messaggio "*keep alive*" e il successivo).

Le uniche eccezioni sono i cambiamenti di stato a "varco occupato" e "controllo online disabilitato", che vengono segnalati solo al primo messaggio "*keep alive*" già schedulato, in quanto l'invio immediato non è necessario. Questo perché il passaggio a "varco occupato" è sempre conseguente ad una transazione valida o ad uno sblocco singolo, che comportano già l'invio di un messaggio immediato: infatti, mentre nel caso offline (come si è visto al §6.2 a pag. 89) non viene comunque registrato nulla fino al completamento del transito o, se questo non avviene, allo scadere di un timeout, nel caso online appena effettuata la transazione valida o lo sblocco singolo viene subito inviato un messaggio del tipo "transazione online"

(vedi §12.1 a pag. 137) a cui viene aggiunta in coda la stringa "**&gate=begin**"; se il "server tag" *$transaction$* è incluso nel parametro **httpOnlineMessage**, come per default , il messaggio inviato contiene anche il record relativo alla transazione nel formato che avrebbe nel caso in cui il transito andasse a buon fine, cioè con i campi **CONTROLLI**="00" e **ESITO**="0" (vedi §7 a pag. 96). Successivamente, al completamento del transito o allo scadere del timeout verrà inviato un secondo messaggio dello stesso tipo, ma con il campo **ESITO** che questa volta può assumere il valore "0" in caso di transito completato oppure "1" in caso di timeout scaduto, e a cui viene aggiunta in coda la stringa "**&gate=end**".

E' anche possibile inviare dei comandi online per forzare un cambiamento di stato del varco da remoto. A tale scopo si può usare la risposta al messaggio "*keep alive*" (va ricordato che il server può "forzare" in qualunque momento l'invio immediato di un pacchetto "*Keep Alive*" proprio allo scopo di eseguire un comando in tempo reale , come spiegato al §12.3 a pag. 140), aggiungendovi il campo:

gateCmd=<*CMD*>

dove <*CMD*> può essere uno dei seguenti comandi:

entry sblocco del varco per una singola entrata: equivale all'attivazione dell'ingresso digitale assegnato tramite il parametro **ManualUnlockIN** (vedi §6.3 a pag. 90). Ha effetto solo se il varco si trova attualmente in stato normale. Genera un immediato messaggio del tipo "transazione online" con in coda la stringa "**&gate=begin**", in cui il "server tag" *$transaction$* (se incluso nel parametro **httpOnlineMessage**, come per default) ha il campo **CODICE_UTENTE** riempito con soli zeri "000..000", **SOURCE**="8" (vedi §7 a pag. 96), **DIREZIONE**="1", **CONTROLLI**="00" e **ESITO**="0". Questo è uno stato temporaneo: al completamento del transito o allo scadere del timeout verrà inviato un secondo messaggio dello stesso tipo, ma con il campo **ESITO** che questa volta può assumere il valore "0" in caso di transito completato oppure "1" in caso di timeout scaduto, e a cui viene aggiunta in coda la stringa "**&gate=end**". Inoltre, a questo seguirà subito dopo un messaggio "*keep alive*" con in coda la stringa "**&gateStatus=H**" a segnalare il ritorno allo stato normale.

exit sblocco del varco per una singola uscita: equivale all'attivazione dell'ingresso digitale assegnato tramite il parametro **ManualUnlockOUT** (vedi §6.3 a pag. 90). Tutto funziona come nel caso del comando **entry** sopra descritto, ma in questo caso il campo **DIREZIONE** vale "0".

emergency sblocco permanente del varco e segnalazione di emergenza (messaggio "**Emergenza**"): equivale all'attivazione dell'ingresso digitale assegnato tramite il parametro **Emergency** (vedi §6.3 a pag. 90). Genera un immediato messaggio "*keep alive*" con in coda la stringa "**&gateStatus=G**". Si tratta di uno stato permanente da cui si può uscire solo inviando un altro comando a scelta fra **operative** (ritorno allo stato normale) oppure **disable** o **free**, oppure attivando e disattivando manualmente l'ingresso digitale assegnato tramite il parametro **Emergency**.

disable blocco continuo del varco (messaggio "**Non disponibile**"): equivale all'attivazione dell'ingresso digitale assegnato tramite il parametro **GateLocked** (vedi §6.3 a pag. 90). Genera un immediato messaggio "*keep alive*" con in coda la stringa "**&gateStatus=C**". Si tratta di uno stato permanente da cui si può uscire solo inviando un altro comando a scelta fra **operative** (ritorno allo stato normale) oppure **emergency** o **free**, oppure attivando e disattivando manualmente l'ingresso digitale assegnato tramite il parametro **GateLocked**.

free sblocco permanente del varco senza segnalazione di emergenza (messaggio "**Varco aperto**"). Genera un immediato messaggio "*keep alive*" con in coda la stringa "**&gateStatus=F**". Si tratta di uno stato permanente da cui si può uscire solo inviando un altro comando a scelta fra **operative** (ritorno allo stato normale) oppure **emergency** o **disable**.

operative ritorno allo stato normale. Ha effetto solo se il varco si trova attualmente in uno dei seguenti stati: disabilitato, libero, emergenza, controllo online disabilitato. In tutti questi casi (tranne l'ultimo) genera un immediato messaggio "*keep alive*" con in coda la stringa "**&gateStatus=H**".

no_ctrl disabilitazione della notifica relativa all'attuale stato del varco: in modalità online non vengono più generati messaggi "*keep alive*" immediati in seguito a qualunque tipo di evento locale o successivo comando da remoto, inoltre i consueti messaggi "*keep alive*" periodici hanno sempre in coda la stringa fissa "**&gateStatus=N**", anche se in realtà in locale il terminale si comporta come al solito in seguito a

ogni tipo di evento o comando, e mostra a video i relativi messaggi. Si tratta di uno stato permanente da cui si può uscire solo inviando il comando online **operative**.
Nota: anche in caso di eventuale passaggio in modalità offline (o server non in linea), se il terminale si trova già in questo stato vi rimane comunque, e come conseguenza non registra nel file TRANSACTIONS.TXT nessun record relativo all'emissione o al rientro di qualunque evento (vedi §7.1 a pag. 100).

7. TRANSAZIONI

Se il parametro **Offline** all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 31) vale '1' (modalità offline), oppure '3' (default: modalità semi-online) ma il server non è in linea o risponde alle transazioni ricevute in tempo reale specificando di salvarle comunque sul terminale (vedi §12.2 a pag. 139), le transazioni vengono registrate in locale nel file di testo TRANSACTIONS.TXT, appositamente ed esclusivamente per lo scarico via FTP, in formato standard oppure personalizzato.

Inoltre vengono anche registrate, questa volta in un formato proprietario e non modificabile, nel file riservato **btransactions.loc** per consentirne (anche nel caso in cui il file TRANSACTIONS.TXT venga scaricato e poi cancellato) la revisione locale oppure la successiva ritrasmissione, record per record, mediante un client HTTP in modalità *batch* (vedi §12 e §12.5 a pag. 144 in particolare). E' anche possibile recuperare tutte le transazioni che sono state in precedenza registrate sul terminale riesportando il contenuto di **btransactions.loc** in un apposito file TRANSACTION_BACKUP.TXT che conterrà tutte le transazioni nello stesso formato con cui vengono registrate nel file TRANSACTIONS.TXT (vedi dettagli più avanti).

Come per tutti gli altri, anche il file TRANSACTIONS.TXT viene salvato nella micro-SD Card rimovibile (tenete a mente che per rimuovere la micro-SD card interna il terminale deve essere smontato e aperto).

Il formato standard dei record di TRANSACTIONS.TXT relativi alle transazioni prevede un numero variabile di campi separati dal carattere virgola ",".

E' anche possibile definire un formato personalizzato dei record (vedi §7.2 a pag. 101): ciò consente di produrre un file di testo TRANSACTIONS.TXT avente già un formato compatibile con un sistema di gestione rilevazione presenze di terze parti (è possibile, ad esempio, usare lo stesso formato già utilizzato in precedenza per l'esportazione dei dati da altri terminali AXESS TMC con il programma TRAXiT32).

Una volta che avete deciso il formato di cui avete bisogno per il file TRANSACTIONS.TXT, potrete calcolare il massimo numero di transazioni che potete registrare localmente sul terminale, dividendo la capacità della micro-SD card (meno i pochi KB necessari per i file di configurazione, come PARAMETERS.TXT) per la dimensione di un singolo record di TRANSACTIONS.TXT (considerate 1 byte per ciascuna cifra o separatore).

I possibili campi dei record di TRANSACTIONS.TXT relativi alle transazioni nel formato standard sono:

AAAAMMGG,HHMMSS,DIREZIONE,COD_CAUSALE,COD_UTENTE,CONTROLLI,ESITO,SORGENTE,EDIZIONE***,**VARCO****,*UTC,ORA_LEGALE*,TRACCIATO_TESSERA**,SLAVE_ID**,LETTORE**,TIPO**,*HASH|PAYLOAD*

(**i campi in grassetto sono sempre presenti**, il campo marcato con * viene riempito solo nei messaggi online – vedi §12.1 a pag. 137, quindi è sempre vuoto nei record memorizzati in locale in TRANSACTIONS.TXT, i campi marcati con ** sono attualmente non gestiti – ***nota***: nelle versioni di fw **a***NN*_**build***nnnn* non erano neppure presenti i relativi separatori "," – e quindi in realtà sono sempre lasciati vuoti, *quelli in corsivo sono opzionali*)

AAAAMMGG data della transazione: AAAA=anno, MM=mese, GG=giorno

HHMMSS ora della transazione: HH=ore nel formato 24h, MM=minuti, SS=secondi

DIREZIONE 0=uscita, 1=entrata

COD_CAUSALE, (default vuoto)

COD_UTENTE codice utente (numerico o alfanumerico) estratto dalla carta secondo i parametri **CardCodeBegin** e **CardCodeLength**, vedi §4.11 a pag. 45).
Nota: Questo campo viene riempito con soli zeri "000..000" in caso di gestione avanzata del varco attiva e sblocco da pulsante manuale o da comando online per singolo transito (vedi §6.3 a pag. 90 e §6.5 a pag. 93).

CONTROLLI campo relativo all'esito della logica di controllo accessi (se attivata), altrimenti fisso a "00".

Nota: tutti i valori di questo campo diversi da "00" possono comparire in un record di TRANSACTIONS.TXT solo se il controllo accessi è attivato (vedi §5 a pag. 68) e se è stata abilitata la registrazione di tutti i tentativi di accesso (inclusi quelli risultati non validi secondo i criteri di controllo accessi) impostando a 1 il parametro **RecordInvalidAccess** nella sezione *[AccessControl]* del file PARAMETERS.TXT (vedi §4.11 a pag. 37) oppure, analogamente, spuntando la checkbox "**Record invalid access**" nella pagina "**Access Control**" del web server HTTP. Per analizzare la causa dei codici di errore qui elencati fate riferimento al §5.13 a pag. 82:

00 = OK
33 = Tessera non valida
34 = Edizione non valida / Causale non valida
35 = Utente disabil.
36 = Tessera disabil.
37 = Utente scaduto / (*)Tessera scaduta
38 = Non autorizzata
39 = Utente fuori orario
41 = Gruppo aut. assente
42 = Timemod assente
44 = Pincode errato
45 = Giorno non valido
51= Verifica biometrica 1:1 fallita / Utente non trovato nell'archivio biometrico
52= Tessera scaduta
53= Mancano entrambe le tabelle CARDS.TXT e CARDRNGE.TXT
54= Manca la tabella AUTHGRP.TXT
55= Manca la tabella AUTH.TXT
56= Manca la tabella TIMEMOD.TXT
59= Codice comune errato

(*) Valori utilizzati nel caso descritto solo per le versioni di firmware precedenti alla a08_build222

ESITO campo relativo all'esito della transazione (se il controllo accessi e/o la gestione avanzata del varco sono abilitati), altrimenti fisso a "0":

0 = transazione completata correttamente
1 = transazione non completata, poiché non consentita (se il campo CONTROLLI è diverso da "00")
oppure
ma solo se è stata attivata la gestione avanzata di un varco (vedi §6 a pag. 88), per transito non effettuato (il varco non è stato aperto nonostante fosse stato sbloccato, in questo caso il campo CONTROLLI="00")

SORGENTE

1= lettore primario (READER 1)
2=lettore secondario (READER 2, eccetto FingerBOX)
3=lettore esterno su morsettiera a vite (EXTERNAL READER) o collegato tramite scheda di espansione opzionale 914 NeoMAX
4= digitazione manuale del codice utente (vedi §4.11 a pag. 31)
5=transazione del tipo "solo PIN"
6=FingerBOX, ma solo se usato in identificazione biometrica 1:N, cioè in modalità "solo dito"
Nota: nel caso di verifica biometrica 1:1 conclusa con successo, il campo SOURCE non assume il valore 6, bensì il valore corrispondente al dispositivo tramite cui è stato inserito il codice tessera che è stato poi sottoposto a verifica biometrica)

EDIZIONE* questo campo viene riempito, opzionalmente, solo nei messaggi online (vedi §12.1 a pag. 137): se il parametro **EditionLength** (vedi §4.7 a pag. 45) nella sezione del file PARAMETERS.TXT relativa al lettore che è stato usato per leggere la carta è diverso da '0', questo campo contiene il codice edizione letto nella carta (1 o 2 cifre), altrimenti viene lasciato vuoto, come risulta sempre essere in tutti i record registrati in locale in TRANSACTIONS.TXT. ***Nota***: in ogni caso il separatore "," che precede questo campo è comunque sempre presente.

VARCO**, TRACCIATO_TESSERA**, SLAVE_ID**, LETTORE**, TIPO** Campi attualmente non gestiti e quindi sempre lasciati vuoti. ***Nota***: in ogni caso i separatori "," che precedono ciascuno di questi campi sono comunque sempre presenti.

Campi opzionali:

UTC

Differenza fra il fuso orario locale e quello universale UTC/GMT (vedi parametro **RecordUTC** nella sezione *[TimeSettings]* del file PARAMETERS.TXT al §4.11, pag. 55):

+HHMM se la differenza è positiva
-HHMM se la differenza è positiva
Z se il terminale si trova nel fuso orario del meridiano di Greenwich

ORA_LEGALE

0= transazione effettuata durante l'orario sorale, 1= ora legale (vedi parametro **RecordDayLightSaving** nella sezione *[TimeSettings]* del file PARAMETERS.TXT al §4.11, pag. 54)

HASH

Questo campo aggiuntivo è presente solo se è stato impostato il parametro **TrnsHash**=1 nella sezione *[System]* del file PARAMETERS.TXT (vedi §4.11 a pag. 53): si tratta di un valore di 32 cifre esadecimali calcolato a partire dai dati essenziali di ciascuna transazione (codice personale, data, ora, direzione di transito e indirizzo MAC del terminale su cui è stata effettuata) mediante un algoritmo non invertibile (**PBKDF2**) e usando come chiave ("*salt data*") una chiave di cifratura scelta a piacere (vedi §17.4 a pag. 163).

PAYLOAD

Questo campo aggiuntivo, sempre preceduto da un carattere separatore "|" *pipe* o chr(124), è presente se il parametro **Payload** nella sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 35) è impostato ad un valore diverso da '0'. In tal caso, il suo contenuto si suppone venga scelto fra un elenco di possibili opzioni; attualmente, comunque, l'unica opzione disponibile è l'intero codice letto a seguito della decodifica della tessera applicata e prima dell'estrazione del codice personale (noto anche come "RAW data"), corrispondente al valore '1' del parametro.

Il file TRANSACTIONS.TXT può essere scaricato via FTP usando un programma client FTP standard. Sono a disposizione, su richiesta, dei semplici file *batch* (*.bat) per effettuare lo scarico dei dati via FTP in maniera automatica da uno o più terminali X1/X2.

È inoltre possibile schedulare degli invii automatici del file TRANSACTIONS.TXT verso un server FTP raggiungibile dal terminale. Per maggiori dettagli si veda il §7.3 a pag. 102.
Il file TRANSACTIONS.TXT può anche essere copiato manualmente su una chiavetta di memoria USB, seguendo le modalità illustrate al §14 pag. 148.

Tutte le transazioni registrate in locale possono comunque essere scaricate, record per record, mediante un server HTTP in modalità *batch* (vedi §12 e §12.5 a pag. 144 in particolare): in questo caso esse vengono prelevate direttamente dal file riservato **btransactions.loc** (che tiene anche nota di quali siano le transazioni ancora "pendenti", cioè non ancora ricevute e/o non confermate dal server), quindi rimangono disponibili anche nel caso in cui il file TRANSACTIONS.TXT sia già stato scaricato e poi cancellato. Anche se il formato di **btransactions.loc** non è modificabile, ciascun messaggio HTTP relativo ad una transazione inviata in modalità *batch* contiene sempre un campo avente lo stesso formato (standard o personalizzato) specificato per i record di TRANSACTIONS.TXT.

Se il parametro **DeleteOld** (vedi §4.11 a pag. 32) è impostato a 1, dopo che il file riservato **btransactions.loc** ha raggiunto una certa dimensione, e se sono soddisfatte le condizioni definite nel seguito basate sul valore del parametro **MaxPendingRecord** (vedi §4.8 a pag. 31), alla successiva transazione effettuata tale file viene automaticamente rinominato in "btransactions.0.loc", e la nuova transazione viene memorizzata all'interno di un nuovo btransactions.loc; allo stesso tempo, l'attuale file "TRANSACTIONS.TXT" viene automaticamente rinominato in "TRANSACTIONS.0.TXT", e la nuova transazione viene memorizzata all'interno di un nuovo TRANSACTIONS.TXT. Successivamente, ogni volta che un nuovo btransactions.loc viene rinominato secondo le stesse modalità, tutti i precedenti file "btransactions.*n*.loc" e "TRANSACTIONS.*n*.TXT", se presenti, vengono a loro volta rinominati rispettivamente in "btransactions.*(n+1)*.loc" e "TRANSACTION.*(n+1)*.TXT", e il file corrente viene sempre rinominato rispettivamente in "btransactions.0.loc" e "TRANSACTIONS.0.TXT". I più vecchi file di transazioni possono essere "btransactions.3.loc" e "TRANSACTIONS.3.TXT", quindi in caso di successivi superamenti del valore massimo tali file vengono automaticamente cancellati. Con questo metodo, qualsiasi transazione nel file TRANSACTIONS.*n*.TXT file è sempre contenuta nel corripondente file btransactions.*n*.loc con lo stesso valore *n* , e il numero di records in btransactions.loc è sempre uguale o maggiore del numero di records in TRANSACTIONS.TXT.

Vediamo ora nel dettaglio in quali situazioni può avvenire la rinomina dei file:

1) In caso di gestione puramente offline (ovvero quando non c'è un server HTTP per scaricare le transazioni, neppure in modalità *batch*), quando il numero totale delle transazioni precedentemente registrate all'interno di **btransactions.loc** (e che ovviamente risultano ancora tutte "pendenti") ha raggiunto il numero specificato dal parametro **MaxPendingRecord**;
2) In caso di gestione con server HTTP, quando il numero totale delle transazioni precedentemente registrate all'interno di **btransactions.loc** ha raggiunto il numero specificato dal parametro **MaxPendingRecord**, sempre che il server HTTP abbia già ricevuto (e confermato) tutte le transazioni già presenti in tale file, ovvero non vi siano transazioni ancora "pendenti";
3) In caso di gestione con server HTTP, se il numero totale delle transazioni precedentemente registrate all'interno di **btransactions.loc** ha raggiunto il numero specificato dal parametro **MaxPendingRecord** ma vi sono delle transazioni ancora "pendenti", la rinomina del file non avviene e quindi la nuova transazione viene regolarmente registrata all'interno del btransactions.loc già presente, la cui dimensione (in record) risulterà quindi essere maggiore del numero specificato dal parametro **MaxPendingRecord**; la stessa cosa avverrà per le transazioni successive, fintanto che il server non avrà ricevuto (e confermato) tutte le transazioni già presenti nel file, ovvero non vi siano più transazioni ancora "pendenti": solo a quel punto, alla prima nuova transazione avverrà la rinomina dei file;
4) In caso di gestione con server HTTP, se il server rimane non raggiungibile per un tempo molto lungo, il numero totale delle transazioni all'interno di **btransactions.loc** può continuare a incrementarsi anche di molto oltre il numero specificato dal parametro **MaxPendingRecord**, almeno fino al punto in cui sarà il numero delle sole transazioni ancora "pendenti" a raggiungere il numero specificato dal parametro **MaxPendingRecord**: solo a quel punto, alla prima nuova transazione avverrà la rinomina dei file(*).

Da quanto detto segue che nel caso in cui un server HTTP ricevesse (e confermasse) costantemente solo una parte delle transazioni ancora "pendenti" (ma non tutte), il numero totale delle transazioni all'interno di btransactions.loc potrebbe continuare a incrementarsi fino a riempire del tutto la memoria disponibile: è necessario pertanto che il server HTTP sia in grado, prima che questo succeda, di ricevere (e confermare) tutte le transazioni ancora "pendenti".

(*)***Attenzione***: quando btransactions.loc viene rinominato, viene anche azzerato il contatore delle transazioni "pendenti": pertanto tutte le transazioni che risultavano essere ancora "pendenti" non verranno in ogni caso più inviate all'eventuale server HTTP, il quale quindi ne perderebbe completamente traccia (anche se rimangono comunque accessibili via FTP all'interno dei file rinominati).

Al contrario, se il parametro **DeleteOld** è impostato a 0, nei casi 1) e 4) descritti in precedenza , ovvero quando il numero delle sole transazioni ancora "pendenti" raggiunge il numero specificato dal parametro **MaxPendingRecord**, il terminale controlla la presenza del file **TRANSACTIONS.TXT**, assumendo implicitamente che si stia effettuando una gestione puramente offline: se tale file è presente, per evitare di bloccare ogni nuova registrazione e al contempo evitare di perdere traccia delle transazioni che potrebbero non essere ancora state scaricate via FTP, la nuova transazione viene regolarmente registrata all'interno del **btransactions.loc** già presente, sforando quindi il limite specificato dal parametro **MaxPendingRecord**; la stessa cosa avverrà per le transazioni successive, fintanto che chi gestisce il terminale non avrà scaricato e cancellato via FTP il file TRANSACTIONS.TXT: solo a quel punto, alla successiva transazione effettuata avverrà la rinomina dei file esattamente come nel caso del parametro **DeleteOld**=1. Se invece il file TRANSACTIONS.TXT non viene mai cancellato, il numero di transazioni ancora "pendenti" continuerà a incrementarsi fino a raggiungere il doppio del numero specificato dal parametro

MaxPendingRecord: solo a quel punto il terminale si rifiuta di registrare ogni nuova transazione, mostrando a video il messaggio di errore “Err. memoria piena”.

Nei casi 2) e 3), invece, ovvero quando il numero totale delle transazioni precedentemente registrate all’interno del file **btransactions.loc** risulta essere maggiore o uguale al numero specificato dal parametro **MaxPendingRecord**, ma nessuna di esse è ancora “pendente” , allora alla successiva transazione effettuata avviene la rinomina dei file esattamente come nel caso del parametro **DeleteOld**=1.

(**)Diversamente dal caso **DeleteOld**=1, poiché i file non vengono rinominati risulta impossibile che l’eventuale server HTTP perda traccia di una qualunque transazione.

Nota: E’ anche possibile recuperare tutte le transazioni che sono state in precedenza registrate sul terminale riesportando il contenuto di tutti i file **btransactions*.loc** ancora presenti in un apposito file TRANSACTION_BACKUP.TXT e nello stesso formato con cui vengono registrate nel file TRANSACTIONS.TXT. A tale scopo è sufficiente usare il pulsante “**Recover**” nella pagina “**System**” del web server HTTP del terminale, o caricare un apposito file via FTP, vedi §16 a pag. 156.

7.1 EMISSIONE E RIENTRO DI EVENTI RELATIVI ALLA GESTIONE AVANZATA DI UN VARCO

Se è stata attivata la gestione avanzata di un varco (vedi §6 a pag. 88) e se X1/X2 è in modalità offline (o comunque se il server non è in linea), il terminale registra nel file TRANSACTIONS.TXT anche altri record oltre a quelli relativi alle transazioni, per tenere traccia anche degli eventi varco (passaggi da uno stato “normale” ad uno stato “anomalo”) che si sono verificati. In particolare, per ciascun tipo di evento viene registrato un primo record subito dopo il passaggio allo stato anomalo (definito “EMISSIONE” dell’evento) ed un secondo record solo una volta che il varco è tornato allo stato normale (definito “RIENTRO” dell’anomalia relativa al medesimo evento). Il formato dei record di TRANSACTIONS.TXT relativi all’emissione o al rientro di un evento non è modificabile e prevede un numero variabile di campi separati dal carattere virgola “,”:

AAAAMMGG,HHMMSS,FASE,,,EVENTO,0,7,,VARCO***,UTC,ORA_LEGALE,,,,,HASH*

(**i campi in grassetto sono sempre presenti**, il campo marcato con ** è attualmente non gestito e quindi in realtà è sempre lasciato vuoto, *quelli in corsivo sono opzionali e hanno lo stesso significato di quelli visti al* §7)

AAAAMMGG data dell’emissione / rientro dell’evento: AAAA=anno, MM=mese, GG=giorno

HHMMSS ora dell’emissione / rientro dell’evento: HH=ore nel formato 24h, MM=minuti, SS=secondi

FASE 5=emissione, 6=rientro

EVENTO tipologia dell’evento:

- 03 = Varco bloccato (blocco manuale continuo): questo stato permane per tutto il tempo di attivazione dell’ingresso digitale assegnato tramite il parametro **GateLocked**, vedi §6.3 a pag. 90
- 07 = Varco in emergenza (sblocco manuale continuo): questo stato permane per tutto il tempo di attivazione dell’ingresso digitale assegnato tramite il parametro **Emergency**,vedi §6.3 a pag. 90
- 11 = Varco forzato (aperto a riposo): questo stato permane fino alla richiusura del varco
- 12 = Varco non richiuso dopo apertura comandata: questo stato permane fino alla richiusura del varco

VARCO** / *UTC* / *ORA_LEGALE* / *HASH* campi opzionali che hanno lo stesso significato già visto al §7 a pag. 96 nel caso delle transazioni

Nota: solo nel caso in cui il terminale sia stato messo nello stato “Notifica stato varco disabilitata” (mediante l’invio di un comando online **gateCmd=no_ctrl**, vedi §6.5 a pag. 93) prima di passare in modalità offline (o server non in linea), allora non registra nel file TRANSACTIONS.TXT nessun record relativo all’emissione o al rientro di qualunque evento.

7.2 DEFINIZIONE DI UN FORMATO PERSONALIZZATO

Per fare in modo che il file di testo TRANSACTIONS.TXT venga creato con i record delle transazioni in un formato personalizzato è necessario impostare il parametro **CustomRecord** all'interno sella sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 32).

Questo parametro, normalmente vuoto, è una stringa che può contenere un numero qualunque di identificatori di campo (la cui sintassi è la stessa usata per i formati di esportazione del programma TRAXiT32), posti in ordine qualunque e intervallati da un numero qualunque di caratteri fissi usati come riempitivi o come separatori di campo. L'unico limite è la lunghezza totale della stringa che non può superare i 68 caratteri.

E' anche possibile inserire una qualunque lettera normalmente riservata agli identificatori di campo (è sufficiente anteporle un carattere di controllo '\'), oppure alcuni caratteri speciali, come la tabulazione (TAB, ASCII 9) ed il ritorno a capo (<CR><LF>).

I possibili identificatori di campo nel formato personalizzato sono:

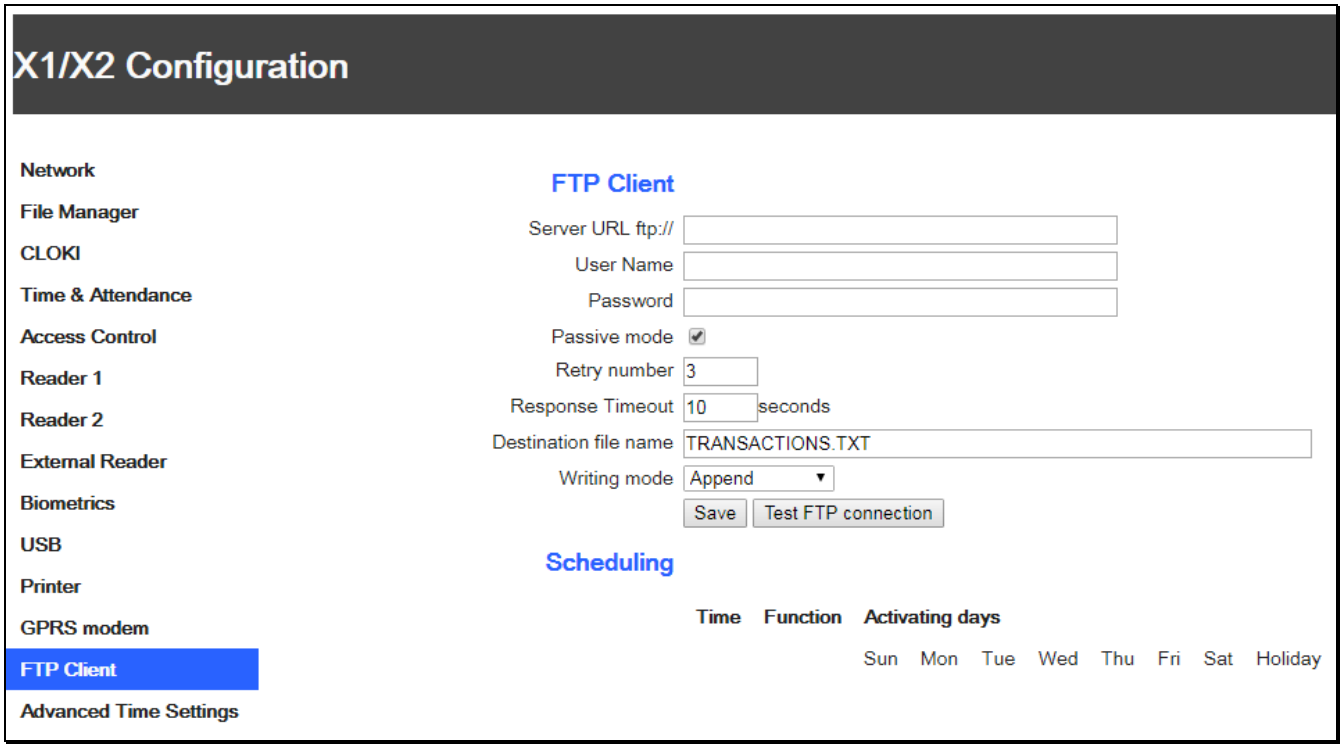
T o TT o TTT	identificatore del terminale (contenuto completo del parametro **TermID**, vedi §4.11 a pag. 56)
YYYY o YY	anno con secolo (es. 2011) o senza secolo (es. 11).
Y	cifra meno significativa dell'anno (0..9).
yy	anno in formato DATING (yy=anno-1980). Es. 2011=31.
MM	mese (01=Gennaio..12=Dicembre).
DD	giorno del mese (01..31).
hh	ore (00..23).
mm	minuti (00..59).
ss	secondi (00..59).
V	direzione di passaggio (default: entrata->"1", uscita->"0"). La stringa inserita può essere personalizzata cambiando il valore dei parametri **CustomEntry** e **CustomExit** all'interno sella sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 32). Nei record relativi ad eventi invece che a transazioni, questo campo è invece sempre fisso a "0".
X..XXX	codice causale. Questo campo può contenere da 1 a 8 segnaposto 'X': se i codici causale specificati nel file REASONS.TXT (vedi §4.4 a pag. 19) sono più lunghi ne verrà registrata solo la parte parte più a destra, se invece sono più corti il campo verrà riempito con zeri a sinistra; in caso di transazione senza causale il campo avrà tutte le cifre fisse a '0'.
C..CCC	codice personale. Se il numero di segnaposto 'C' (variabile da 1 a 20) è inferiore al valore del parametro **CardCodeLength** (vedi §4.11 a pag. 45) verrà registrata solo la parte più a destra del codice utente letto, se invece è maggiore il campo verrà riempito con zeri a sinistra. Nei record relativi ad eventi invece che a transazioni, questo campo viene interamente riempito con tanti spazi vuoti " " quanti sono i segnaposto utilizzati.
c..ccc	funziona come il campo C..CCC, ma in caso sia necessario un riempimento vengono inseriti degli spazi (" ") a destra invece che gli zeri a sinistra.
S	codice "sorgente" della transazione (è equivalente al campo **SORGENTE** descritto al §7, pag. 96).

	Nei record relativi ad eventi invece che a transazioni, questo campo è sempre uguale a “7”.
E	Esito finale della transazione (se è abilitato il controllo di accessi e/o la gestione avanzata del varco), altrimenti fisso a "0": 0 = Transazione completata, 1 = Transazione non completata (è equivalente al campo **ESITO** descritto al §7 a pag. 96). Nei record relativi ad eventi invece che a transazioni, questo campo è sempre uguale a “0”.
ee	Esito della logica di controllo accessi (se abilitata), altrimenti fisso a “00” (è equivalente al campo **CONTROLLI** descritto al §7 a pag. 96, ed è sempre espresso su 2 cifre). Nei record relativi ad eventi invece che a transazioni, è invece equivalente al campo **EVENTO** descritto al 7.1 a pag. 100, sempre espresso su 2 cifre.
Z	Campo opzionale “valore di hash” della transazione/evento (è equivalente al campo *HASH*, presente solo nel caso descritto al §7 a pag. 96, ed è sempre costituito da 32 cifre esadecimali).
PP	Campo opzionale “payload esteso” (è equivalente al campo *PAYLOAD*, presente solo nel caso descritto al §7 a pag. 96). Il segnaposto ‘PP’ (specificare sempre 2 caratteri maiuscoli ‘P’ adiacenti) viene effettivamente sostituito dall’intero contenuto di tale campo soltanto in quel caso, altrimenti appare invariato nei record delle transazioni personalizzate. Nei record relativi ad eventi invece che a transazioni, il segnaposto PP appare sempre invariato.
\c	inserisce un carattere generico “*c*”, anche se normalmente riservato ad un identificatore di campo
^	inserisce un carattere TAB (chr(9)).
|	“*pipe*” o chr(124): inserisce un ritorno a capo nel file, costituito dai caratteri <CR> e <LF>.

7.3 INVIO DELLE TRANSAZIONI TRAMITE CLIENT FTP

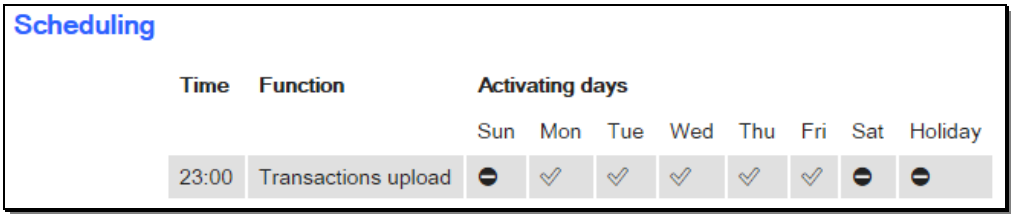
A partire dalla versione di firmware a08_build018, il terminale include un client FTP che permette il caricamento automatico del file TRANSACTIONS.TXT corrente su un server FTP raggiungibile dal terminale.
Le opzioni di connessione del client FTP possono essere configurate editando direttamente il file PARAMETERS.TXT come indicato al §4.11 a pag. 58 , oppure collegandosi tramite web browser al terminale, come indicato al §4 pag. 14.

Se si utilizza questo secondo metodo, selezionando nel menù di sinistra della pagina web la voce “FTP Client”, sarà visualizzata la seguente schermata. Qui è possibile inserire l’indirizzo del server a cui connettersi, le credenziali di autenticazione, il numero di tentativi in caso di fallimento, il timeout di risposta per ciascun tentativo e le regole di creazione del file destinazione all’interno del server FTP, specificandone il nome e la modalità di scrittura. Ciascun parametro è descritto nei dettagli al §4.11 a pag. 58.



Dopo aver salvato le impostazioni, è possibile premere il pulsante "Test FTP connection" per verificarne la correttezza. Se il test viene completato con successo, nella stessa posizione sul server dove verrà salvato il file destinazione, sarà presente un file di testo "_FTP_TEST.TXT" creato utilizzando le stesse modalità di invio che verranno utilizzate durante le esportazioni schedulate.

La schedulazione delle esportazioni deve essere configurata manualmente editando il file ALARMS.TXT come indicato al §4.2 a pag. 18. La stessa cosa si può anche fare mediante il link "**TABELLE / ALLARMI**" nel componente opzionale "**CLOKI**" (in precedenza denominato "Web Table Editor") se presente (vedi §5.14 a pag. 84). Una volta fatto questo, ricaricando la pagina web "FTP Client" vedrete la tabella delle schedulazioni aggiornata:



Vediamo adesso come viene gestito il file TRANSACTIONS.TXT sul terminale in caso di utilizzo del client FTP. Allo scopo di evitare l'invio multiplo delle stesse transazioni, all'orario schedulato per l'esportazione il file TRANSACTIONS.TXT (se presente sul terminale) viene immediatamente rinominato nel file temporaneo TRANSACTIONS.FTP. A questo punto il client FTP tenta l'invio al server, e solo in caso di successo rinomina ulteriormente il file in TRANSACTIONS.0.FTP, che verrà mantenuto come backup. In questa maniera il file TRANSACTIONS.TXT verrà ricreato alla prima nuova transazione effettuata, e alla successiva esportazione verranno inviate solo le nuove transazioni: il processo è identico, ma in caso di successo, essendo già presente un file TRANSACTION.0.FTP, questo verrà rinominato in TRANSACTIONS.1.FTP, ed il file temporaneo TRANSACTIONS.FTP appena trasmesso verrà rinominato in TRANSACTIONS.0.FTP. Stessa cosa per l'esportazione seguente, allorché TRANSACTIONS.1.FTP verrà rinominato in TRANSACTIONS.2.FTP, TRANSACTIONS.0.FTP in TRANSACTIONS.1.FTP e TRANSACTIONS.FTP in TRANSACTIONS.0.FTP. A partire da questo momento, e per ogni successiva esportazione, il file più vecchio (TRANSACTIONS.2.FTP) verrà cancellato prima di rinominare gli altri file di backup, che quindi in totale saranno sempre al massimo 3.

Nel caso in cui l'invio al server fallisca per tutti i tentativi previsti, il file temporaneo TRANSACTIONS.FTP non viene rinominato (e quindi neppure gli altri). Alla successiva esportazione schedulata, verrà per prima cosa ritentato l'invio di

questo file: se anche in questo caso l'operazione dovesse fallire, il file TRANSACTIONS.TXT corrente non verrà rinominato, e quindi continuerà ad accumulare transazioni "pendenti" fino alla successiva schedulazione.

Note: il funzionamento del client FTP è ottimizzato. Se il file TRANSACTIONS.TXT non è presente (il che significa che non ci sono nuove transazioni da caricare), la connessione da client FTP non ha neppure luogo.

8. LINGUE

Al primo riavvio del terminale dopo avere formattato la micro-SD, un file di testo ASCII chiamato **LANGUAGE.TXT** viene creato automaticamente. Include tutte le stringhe per i messaggi nelle lingue inglese, italiano, spagnolo, francese e tedesco.

Se volete selezionare una lingua diversa, dovete prima fare in modo che i relativi messaggi siano inclusi nel file LANGUAGE.TXT. Per aggiungere nuovo lingue, o per cambiare i messaggi standard usati dal terminale, è sufficiente caricare un nuovo LANGUAGE.TXT sul file system del terminale via FTP.

Il formato del file LANGUAGE.TXT è il seguente:

[Identificatore Lingua]
Numero messaggio=Messaggio

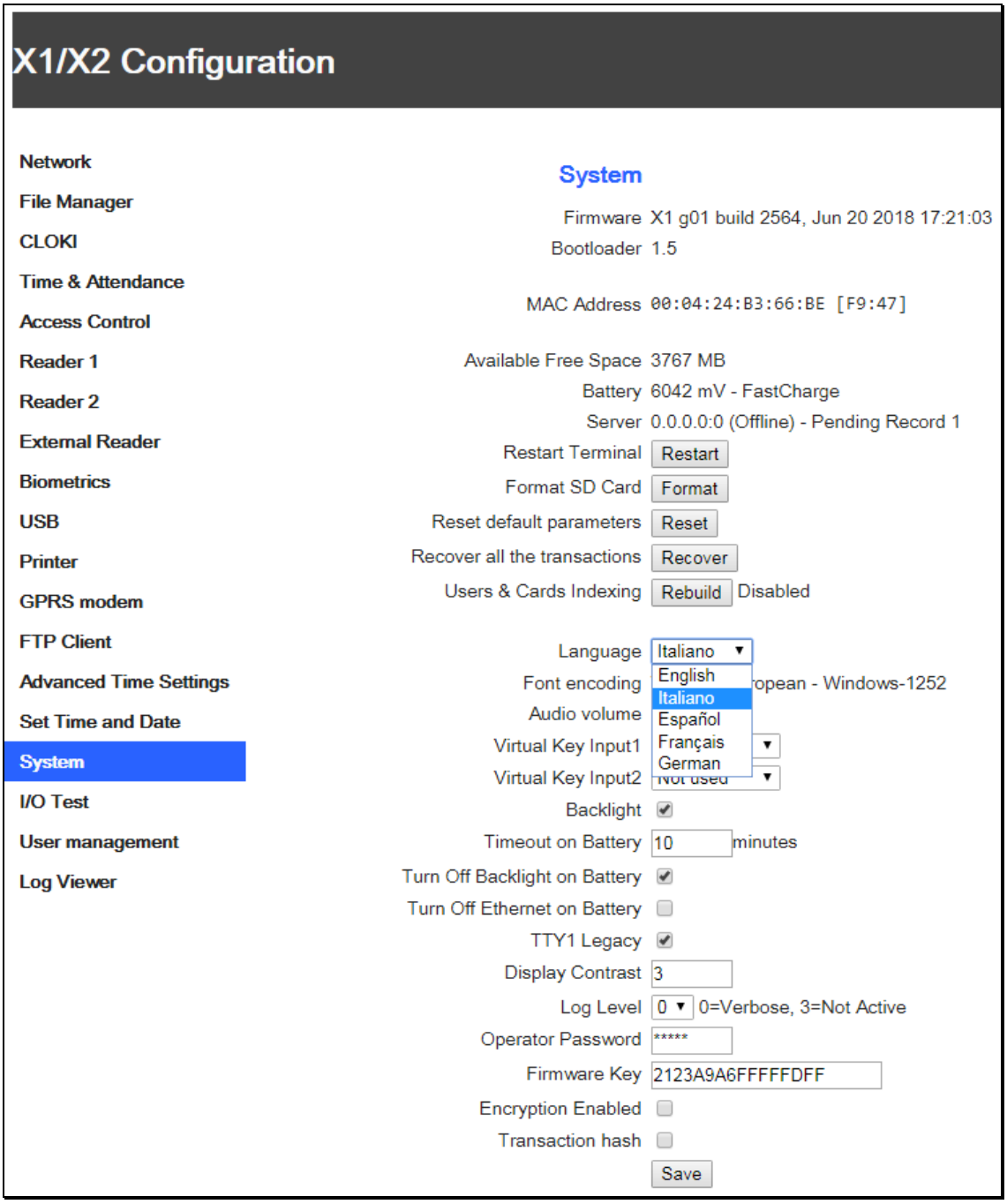
Non c'è limite al numero di lingue che il terminale può gestire: aggiungete tutte le lingue che volete al file LANGUAGE.TXT.

Il file LANGUAGE.TXT non viene controllato periodicamente da X1/X2, per cui dopo aver caricato il nuovo file è necessario riavviare il terminale per rendere effettivi i cambiamenti.

La lingua che verrà effettivamente selezionata, fra quelle presenti nel file LANGUAGE.TXT, è determinate dal valore del parametro **Language** all'interno della sezione *[System]* del file PARAMETERS.TXT (vedi §4.11 a pag. 51), il cui valore di default è "English". Se l'attuale valore del parametro **Language** non viene trovato fra gli "Identificatori di Lingua" presenti nel file LANGUAGE.TXT (ad esempio perché avete appena caricato un nuovo LANGUAGE.TXT che non include la lingua precedentemente impostata, ma non avete ancora cambiato il file PARAMETERS.TXT), allora i messaggi vengono prelevati direttamente dal FW del terminale e mostrati nella lingua di default (inglese).

La lingua può anche essere impostata dalla pagina "**System**" del web server HTTP del terminale, dove viene mostrato un menu a tendina contenente tutti gli "Identificatori di Lingua" trovati nel file in LANGUAGE.TXT. Scorrete la lista e selezionate l'identificatore di lingua desiderato, quindi fate click su "Save".

Nota: nelle future versioni di firmware, altre lingue potrebbero aggiungersi alla lista di quelle disponibili; tenete però presente che l'aggiornamento del firmware non comporta la sostituzione automatica del file LANGUAGE.TXT già presente, pertanto potrebbe essere necessario cancellare tale file e riavviare il terminale per farlo ricreare con le stringhe aggiornate.


X1/X2 Configuration
Network
File Manager
CLOKI
Time & Attendance
Access Control
Reader 1
Reader 2
External Reader
Biometrics
USB
Printer
GPRS modem
FTP Client
Advanced Time Settings
Set Time and Date
System
I/O Test
User management
Log Viewer
System
Firmware X1 g01 build 2564, Jun 20 2018 17:21:03
Bootloader 1.5
MAC Address 00:04:24:B3:66:BE [F9:47]
Available Free Space 3767 MB
Battery 6042 mV - FastCharge
Server 0.0.0.0:0 (Offline) - Pending Record 1
Restart Terminal Restart
Format SD Card Format
Reset default parameters Reset
Recover all the transactions Recover
Users & Cards Indexing Rebuild Disabled
Language Italiano
English
Italiano
Español
Français
German
Font encoding
opean - Windows-1252
Audio volume
Virtual Key Input1
Virtual Key Input2
Backlight
Timeout on Battery 10 minutes
Turn Off Backlight on Battery
Turn Off Ethernet on Battery
TTY1 Legacy
Display Contrast 3
Log Level 0 0=Verbose, 3=Not Active
Operator Password *****
Firmware Key 2123A9A6FFFFFDFF
Encryption Enabled
Transaction hash
Save

9. AGGIORNAMENTO FIRMWARE

Gli aggiornamenti firmware di X1/X2 sono disponibili come file chiamati "**XONE_***VNN***_build***nnn***.bin**".

Le nuove versioni di firmware saranno disponibili per il download nella nostra area partners.

Per aggiornare il firmware, è sufficiente copiare il file "**XONE_***VNN***_build***nnn***.bin**" contenente la nuova versione di firmware nella *root* del file system del terminale via FTP, come fareste con qualunque altro file.

Il terminale riconoscerà automaticamente la presenza del file ed effettuerà la copia del firmware nella memoria flash interna, poi cancellerà il file utilizzato dalla *root* della micro-SD. Dopo un breve tempo il terminale si riavvierà con il nuovo firmware.

L'intero processo (trasmissione FTP inclusa) richiede meno di 10 secondi.

Un qualunque programma client FTP client (ad esempio FileZilla) può essere usato per effettuare la procedura di aggiornamento firmware. **Tuttavia, il componente aggiuntivo di FireFox "FireFTP" è SCONSIGLIATO.**

E' anche possibile aggiornare il firmware in locale mediante una chiavetta USB, come descritto al §14.4 a pag. 150 (utile per terminali *stand-alone* non collegati in Ethernet, o su cui comunque non sia possibile caricare il firmware via FTP).

9.1 AGGIORNAMENTO DEL FIRMWARE DEI LETTORI

Alcuni lettori collegabili a X1/X2 come lettori di "console" (cioè collegati direttamente alla scheda di X1/X2: solo lettori RFID4 e RFID5, ad esempio il modulo interno a doppia tecnologia e doppia testa di lettura RF5-ECO fornito su tutti i modelli di X1/X2 con p/n **930.***xxx***.6***x*, lettori esterni RFID4 con p/n **904.40***x***.14** o lettori esterni RFID5 con p/n **904.50***x***.61**) sono dotati di un firmware aggiornabile direttamente attraverso il file system del terminale.

E' sufficiente copiare il relativo file con estensione **.bin** contenente la nuova versione di firmware (rivolgetevi a Zucchetti AXESS per ottenere il file specifico per un determinato lettore e per stabilire se l'aggiornamento sia in effetti necessario) nella *root* della micro-SD via FTP, come fareste con qualunque altro file.

Dal momento che i lettori di "console" RFID4 e RF5/RFID5 non vengono automaticamente rilevati e univocamente identificati (anche in considerazione del fatto che sono lettori multi-interfaccia), è inoltre necessario che siano stati precedentemente configurati in una delle modalità seriali supportate (parametro **CardDecode**=30 / 32 / 42 / 43 nella corrispondente sezione del file PARAMETERS.TXT): solo in questo caso, infatti, compare un apposito pulsante "**Update FW reader**" nella corrispondente pagina **Reader 1**, **Reader 2** o **External Reader** del web server HTTP del terminale; una volta premuto tale pulsante e dato l'OK alla richiesta di conferma per procedere con l'aggiornamento, il terminale emette una lunga serie di beep ravvicinati che rappresentano un feedback sonoro dell'invio del codice, fino al termine dell'operazione che ha luogo dopo diversi secondi. In questo caso il file con il firmware non viene cancellato, poiché potrebbe essere necessario aggiornare più di un lettore. Se il lettore ha un firmware molto vecchio potrebbe essere necessario riavviarlo manualmente: a tale scopo è sufficiente premere il tasto "**Save**" nella stessa pagina web.

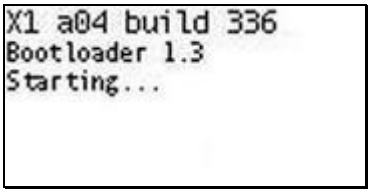
Per verificare la versione di firmware attualmente caricata sui lettori di "console" RFID4 e RF5/RFID5, sempre ammesso che siano stati precedentemente configurati in una delle modalità seriali supportate (parametro **CardDecode**=30 / 32 / 42 / 43 nella corrispondente sezione del file PARAMETERS.TXT), potete inviare un comando costituito da una singola lettera minuscola "v", da inserire nella casella di testo "**Command**" che in tal caso compare nella corrispondente pagina **Reader 1**, **Reader 2** o **External Reader** del web server HTTP del terminale. Attendete quindi che la pagina si ricarichi automaticamente: sotto la casella di testo comparirà la risposta del lettore con la stringa identificativa della versione di firmware.

10. INTERFACCIA UTENTE DI X1/X2

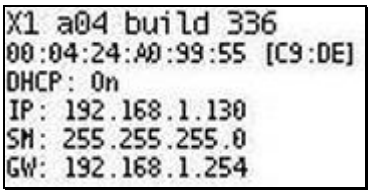
10.1 AVVIO

Un terminale correttamente alimentato si accende da solo. L'avvio del sistema richiede circa 6 secondi. Per riavviare X1/X2 tenere premuto il pulsante [↔] per circa 6 secondi, o premere brevemente il pulsante RESET sulla scheda (vedi §3.2 a pag. 9). Il pulsante [↔] permette anche di spegnere il terminale, ma solo se sta funzionando a batteria. Se si spegne durante il funzionamento a batteria, X1/X2 può essere riacceso premendo un qualunque tasto per almeno 1 secondo.

All'accensione lo schermo mostra prima le versioni di firmware e Bootloader, per 3 secondi:



e poi la configurazione Ethernet, ancora per 3 secondi:



Alla fine vedrete la schermata principale di stand-by, il cui aspetto dipende dalle impostazioni correnti di data e ora e icone di direzione.

10.2 STATO DI ATTESA (PRONTO AD ACCETTARE TRANSAZIONI)

L'aspetto dello schermo può essere cambiato agendo sui seguenti parametri: **DirMode, CompanyName, SecondsShown, AmPm, MonthDay, DateSeparator.**

DirMode = 4
CompanyName = ""
SecondsShown = 1
AmPm = 0
MonthDay = 0
DateSeparator = 47 ("/")

DirMode = 3
CompanyName = ""
SecondsShown = 0
AmPm = 1
MonthDay = 1
DateSeparator = 46 (".")

E' anche possibile personalizzare l'aspetto della schermata principale caricando delle icone di direzione da mostrare al posto di quelle standard, oppure un logo aziendale da mostrare al centro, subito sotto l'orario. I file devono essere caricati nella *root* del terminale, e devono necessariamente chiamarsi **ENTRY.BMP**, **EXIT.BMP** e **LOGO.BMP**, rispettivamente per le icone relative all'entrata e all'uscita, e per il logo aziendale. Tutti questi file devono essere in formato bitmap monocromatico (a 2 colori); considerato che la larghezza totale del display è pari a 128 pixels, le dimensioni massime (L x A) sono 64x20 pixels per le icone e 128x20 per il logo aziendale. Se si desidera mostrare contemporaneamente le due icone di direzione ai lati (**DirMode**=4/5) ed il logo al centro, tuttavia, la somma delle larghezze delle tre bitmap non deve essere superiore a 128 pixels per evitare fastidiose sovrapposizioni. Esempi:

DirMode = 5

File ENTRY.BMP e EXIT.BMP

DirMode = 0/6

File LOGO.BMP

Note:

1) se il file non era presente, appena lo si carica e viene rilevato dal terminale, la relativa immagine viene visualizzata
2) se il file era già presente e ne viene caricato un altro con lo stesso nome, la nuova immagine viene visualizzata solo al riavvio, o dopo aun aggiornamento di configurazione via web server HTTP
3) se **DirMode**=3, l'eventuale logo aziendale non viene comunque visualizzato per lasciare posto alla singola icona della direzione corrente al centro (standard o personalizzata)
4) se **DirMode**=0/6, l'icona relativa alla direzione fissa impostata (standard o personalizzata) viene mostrata solo in assenza di un logo aziendale e solo se il parametro **CompanyName** è vuoto
5) Se è stato caricato un logo aziendale, anche se è stato impostato **CompanyName** con una stringa non vuota, questa stringa non viene comunque visualizzata
6) Se non è stato caricato un logo aziendale, ma è stato impostato un **CompanyName** troppo lungo, tale da sovrapporsi alle icone di direzione standard, né queste ultime né le eventuali icone personalizzate vengono visualizzate

I pulsanti attivi nello stato di attesa "pronto ad accettare transazioni" sono i seguenti (***Nota***: dalla versione di firmware a07_build832 tutti i caratteri alfanumerici vengono accettati all'interno dei codici utente in seguito a lettura di carte per le transazioni di rilevazione presenze / controllo accessi):

.F + ▲ → accesso al menu supervisore

▲ → revisione dati locale (abilitata per default ma disattivabile su necessità, vedi nota al §10.7 a pag. 118)

▼ → menu dei codici causale, solo se è stato precedentemente caricato un file di testo chiamato REASONS.TXT (vedi §4.4 alla pag. 19), o in alternativa un altro file chiamato AXREASON.TXT (vedi §5.10 alla pag. 79)

[<-]-> → inversione direzione di transito (solo se il parametro DirMode all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT, vedi §4.11 a pag.30, è impostato a 3)

oppure

se tenuto premuto per circa 6 secondi, riavvio (se il terminale è alimentato) o spegnimento (se sta funzionando a batteria)

0,1..9 (solo sui modelli X2 con tastiera numerica) →

digitazione manuale di un codice tessera, solo se il parametro **AllowTypeCode** all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag.31) è impostato a 1, o selezionando la checkbox "**Allow Typed Code**" nella pagina "**Time & Attendance**" del web server HTTP.
Nota 1: è possibile inserire un numero di cifre minore o uguale al valore del parametro **CardCodeLength** all'interno della sezione *[Reader1]* del file PARAMETERS.TXT. Se il numero di cifre è minore, il codice verrà completato con riempimento di zeri a sinistra.
Nota 2: se il parametro **HideTypedCode** all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi pag. 34) è stato impostato a 1 (default 0), il codice digitato viene mascherato con asterischi, per evitare che venga visto e successivamente utilizzato da altri utenti non autorizzati.

oppure

digitazione manuale di un codice PIN, se è attivata la modalità "solo PIN" (vedi §5.12 a pag. 81). L'eventuale attivazione della modalità "solo PIN" è prioritaria rispetto all'attivazione della modalità "digitazione manuale di un codice tessera": se sono attive entrambe, alla pressione di un qualunque tasto si passa direttamente all'introduzione del PIN, vedi §10.4 a pag. 112 (non è possibile effettuare manualmente transazioni relative a utenti non associati ad un PIN).

oppure

selezione diretta di una causale, solo se è stato caricato il file FKEY.TXT (vedi §4.5 a pag. 20) e almeno uno fra REASONS.TXT (vedi §4.4 alla pag. 19) e AXREASON.TXT (vedi §5.10 alla pag. 79), e se entrambe le modalità "digitazione manuale di un codice tessera" e "solo PIN" sopra descritte sono disabilitate (queste ultime sono più prioritarie)

Clr → cancellazione di una cifra digitata in precedenza (solo sui modelli X2 con tastiera numerica)

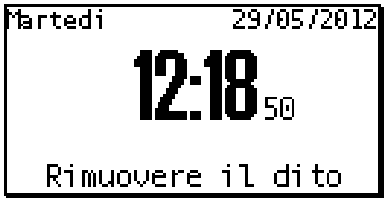
↵ → menu "ridotto" per la selezione delle sole causali e/o delle *enquiries* remote (queste ultime disponibili solo se X1/X2 viene gestito dal programma Xatl@s) associate a tasti numerici per la selezione diretta (scelta rapida), solo se è stato precedentemente caricato il file FKEY.TXT (vedi §4.5 alla pag. 20)

AUTENTICAZIONE BIOMETRICA

Se X1/X2 è equipaggiato con un modulo biometrico esterno FingerBOX per la scansione di impronte digitali, ed è stata attivata la gestione del FingerBOX da parte del terminale come descritto al §3.6 a pag. 11, è possibile usare le impronte digitali per effettuare l'autenticazione biometrica dell'utente. A tale scopo si può procedere in 2 modi diversi:

- **IDENTIFICAZIONE 1:N**

Se è stata abilitata la modalità "autoscan" (come descritto al §11 a pag. 121), il sensore di impronte rimane sempre in uno stato di attesa scansione, pertanto è possibile appoggiare direttamente il dito sul sensore in qualunque momento e procedere all'identificazione dell'utente sulla base del confronto dell'impronta appena scansionata con tutte quelli già registrate nel modulo. Non appena il sensore rileva la presenza del dito, X1/X2 mostra il seguente messaggio:



Se l'utente viene identificato ed il relativo codice tessera è valido, tutto procede come nel caso della lettura di una carta in assenza del modulo FingerBOX (vedi §10.3 qui di seguito), altrimenti compare il messaggio di errore "**Utente non trovato**".

- **VERIFICA 1:1**

Con questa opzione, che è sempre disponibile (anche nella modalità "autoscan"), l'utente deve prima identificarsi mediante la lettura di una tessera o (alle condizioni già descritte in precedenza) la digitazione manuale del codice tessera. A seconda del tipo di tessera utilizzata X1/X2 si comporta diversamente:

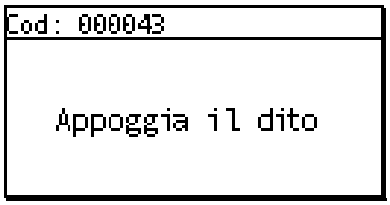
1) se si tratta di una carta Mifare contenente un'impronta, e se il parametro **TemplateSource** all'interno della sezione *[Biometric]* del file PARAMETERS.TXT (vedi §4.11 a pag. 49) è diverso da '1', il terminale legge tutti i dati dell'impronta sulla carta. Questo processo richiede alcuni istanti, durante i quali il terminale mostra il seguente messaggio:

In questo stato non si deve allontanare la carta dal lettore, altrimenti la lettura viene abortita e compare il messaggio "**Tessera persa**". Una volta completata la lettura X1/X2 passa direttamente alla richiesta di scansione del dito (vedi dopo).

Nota: vengono correttamente lette anche le carte (da 1KB o da 4KB) contenenti dei *template* registrati nel formato non-standard a 256 byte (che possono essere

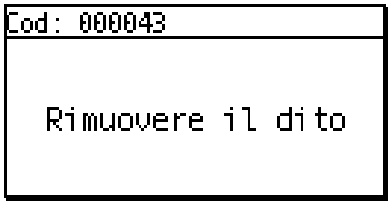
stati salvati solo da terminali 962 SuperTRAX opportunamente configurati).

2) negli altri casi controlla che sia stata già stata registrata almeno un'impronta per il codice tessera introdotto, verificandone la presenza all'interno dei record del file USERCODS.TXT: se non lo trova mostra il messaggio di errore "**Utente non trovato**"(*) e abortisce la transazione, altrimenti pone il sensore in stato di attesa scansione e chiede all'utente di appoggiare il dito per poi effettuare la verifica di identità (a meno che l'utente, oppure tutte le letture effettuate sul lettore di tessere utilizzato, non siano stati esentati dalla verifica biometrica, con le modalità descritte al §11.1 a pag. 125 o al §11.4 a pag. 135), confrontando l'impronta appena scansionata solo con quelle già registrate per il codice tessera introdotto:



Solo se sono stati caricati i file CARDS.TXT (§5.4 a pag. 71) e USERS.TXT (§5.9 a pag. 77), invece del codice della tessera X1/X2 visualizza il nome dell'utente ad esso relativo:

Se entro 10 secondi non viene rilevata la presenza del dito, X1/X2 abortisce la transazione e mostra il messaggio di errore "**Operazione annullata**", altrimenti procede con la verifica di identità e mostra il seguente messaggio:



Entro un secondo, X1/X2 emette il responso: se l'identità dell'utente è stata verificata con successo ed il relativo codice tessera è valido, tutto procede come nel caso della lettura di una carta in assenza del modulo FingerBOX (vedi §10.3 qui di seguito), altrimenti compare il messaggio di errore "**Errore impronta**".

(*) Nota: nel caso in cui sia presente un file CARDS.TXT in formato "esteso" (cioè con tutti i record di 71 caratteri invece che 69 come nella versione standard) contenente un record relativo al codice tessera introdotto con l'apposito flag **B** (biometrico) a '1', il messaggio di errore sarà "**Utente non enrollato**". In tal caso, infatti, si tratta di un codice conosciuto e per il quale è già stato previsto l'utilizzo della biometria, anche se non si è ancora proceduto alla registrazione delle relative impronte. Per ulteriori dettagli si veda il §5.4 a pag. 71.

10.3 DOPO UNA LETTURA DI CARTA, DIGITAZIONE DI CODICE O AUTENTICAZIONE BIOMETRICA

In caso di identificazione biometrica 1:N, o di lettura (o digitazione, vedi paragrafo precedente) di un codice valido (ed eventuale verifica biometrica 1:1), e secondo le impostazioni di **DirMode**:

Nota: a seconda di quale sia il lettore che ha generato la lettura, è possibile fare in modo che venga visualizzata solo una parte del codice personale già estratto dalla tessera, agendo sui parametri **ShowCardCodeBegin** e **ShowCardCodeLength** all'interno delle sezioni *[Reader1]*, *[Reader2]* e *[ExtReader]* del file PARAMETERS.TXT (vedi a pag. 45) relative a ciascun lettore. Anche se viene visualizzata solo una parte del codice personale, nel file TRANSACTIONS.TXT viene sempre memorizzato l'intero codice personale, cioè tante cifre quante sono quelle impostate dal parametro **CardCodeLength** all'interno della stessa sezione del file PARAMETERS.TXT (vedi a pag. 45).

Solo se il controllo degli accessi è stato attivato (vedi §5 a pag. 68), e se sono stati caricati i file CARDS.TXT (§5.4 a pag. 71) e USERS.TXT (§5.9 a pag. 77), in caso di transazione accettata è possibile visualizzare, invece del codice della tessera, il nome dell'utente ad esso relativo:

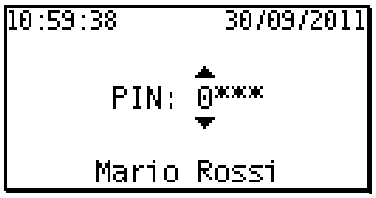
Nota: se il parametro **HideTypedCode** all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi pag. 34) è stato impostato a 1 (default 0) per mascherare la digitazione manuale del codice personale, quest'ultimo non viene neanche mostrato per conferma in caso di transazione accettata (come invece succede normalmente), e neppure in caso di codice inserito mediante lettura di tessera o identificazione biometrica (a prescindere dal valore del parametro **AllowTypeCode**), ma viene comunque mostrato il nome dell'utente relativo a quel codice se sono soddisfatte le condizioni appena descritte.

Il terminale emette suoni politonali diversi nei due casi di transazione valida oppure non valida. In ciascun caso è possibile sostituire il suono di default con un numero di brevi "beep" monotonali variabile da 0 (nessun suono) a 9, agendo sui parametri **BeepOk** e **BeepError** all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi a pag. 30).

E' anche possibile personalizzare il messaggio mostrato a display in seguito all'inserimento di un codice, sia per le transazioni accettate che per quelle rifiutate, impostando rispettivamente i parametri **ScreenOk** e **ScreenError** all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi a pag. 33).

10.4 RICHIESTA CODICE PIN

Se il controllo degli accessi è stato attivato (vedi §5 a pag. 68) e sono stati caricati i file CARDS.TXT (§5.4 a pag. 71) e USERS.TXT (§5.9 a pag. 77), e se il parametro **AskPin** all'interno della sezione *[AccessControl]* del file PARAMETERS.TXT (vedi §4.11 a pag. 37) è stato impostato a 1 (default 0), in caso di lettura di un codice tessera relativo ad un utente per il quale è prevista l'introduzione di un PIN di sicurezza (ma è anche possibile disabilitare la richiesta del PIN in base alla provenienza della lettura di tessera, vedi parametro **DisableFunctions** al §4.11 a pag. 46), il display del terminale si modifica come segue: l'orario non è più visualizzato nel grande formato al centro dello schermo, ma compare in alto a sinistra al posto del giorno della settimana; nella parte bassa compare il nome dell'utente (contenuto nel file USERS.TXT assieme al PIN atteso), mentre al centro compare il prompt di richiesta PIN.

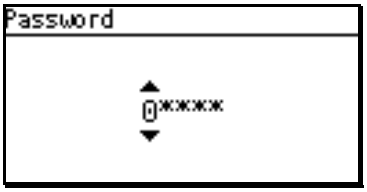


il PIN è parzialmente mascherato per evitare che venga visto da estranei, ma mentre tutte le altre cifre vengono visualizzate come asterischi, quella attualmente in corso di inserimento viene lasciata in chiaro: potete quindi usare i tasti ▲▼ per modificare ciascuna singola cifra (solo su X2, potete anche premere il tasto corrispondente sulla tastiera numerica), ↵ (Enter) per passare alla cifra successiva o per confermare quando vi trovate sull'ultima e **Clr** per tornare indietro di una posizione o abortire quando vi trovate sulla prima. Nota: dopo 10 secondi di inattività, X1/X2 abortisce la transazione mostrando il messaggio "Operazione annullata".

Il prompt di richiesta PIN compare anche (ma questa volta senza il nome dell'utente prelevato da USERS.TXT) quando un server risponde ad una transazione con la richiesta di introduzione del PIN online (vedi §12.2 a pag. 139), o quando viene premuto un qualsiasi tasto numerico (solo su X2) nello stato di attesa lettura carta, se è stata attivata la modalità "solo PIN" (vedi §10.2 a pag. 108).

10.5 MENU SUPERVISORE

Premete **.F** + ▲ per visualizzare il prompt di richiesta password operatore:



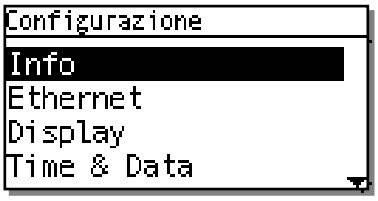
La password è parzialmente mascherata per evitare che venga vista da estranei, ma mentre tutte le altre cifre vengono visualizzate come asterischi, quella attualmente in corso di inserimento viene lasciata in chiaro. Il primo valore mostrato per ciascuna cifra è sempre '0', per cui essendo "00000" la password di default è sufficiente premere ↵ (Enter) 5 volte per accedere al menu supervisore.

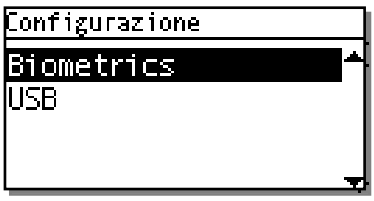
Se avete precedentemente cambiato la password, usate i tasti ▲▼ per modificare una singola cifra (solo su X2, potete anche premere il tasto corrispondente sulla tastiera numerica), ↵ (Enter) per passare alla cifra successiva o per confermare quando vi trovate sull'ultima e **Clr** per tornare indietro di una posizione o abortire quando vi trovate sulla prima.

Nel caso in cui si utilizzi un modulo biometrico esterno FingerBOX per la scansione di impronte digitali è possibile cambiare le modalità di accesso al menu supervisore, mediante la definizione di utenti con funzioni di "amministratore" e usando l'autenticazione biometrica per consentire l'accesso: si veda al proposito il §11.1 a pag. 131. In particolare, ricordate che nel caso in cui sia stato definito almeno un amministratore di tipo "generale" non è più possibile inserire la password manualmente da tastiera, anche se corretta (al momento della conferma il prompt di richiesta password viene sistematicamente abortito e il terminale emette 3 beep consecutivi, oltre a registrare un messaggio di errore nel file LOG.TXT).

Nota: dopo 30 secondi di inattività, X1/X2 esce automaticamente dal menu supervisore.

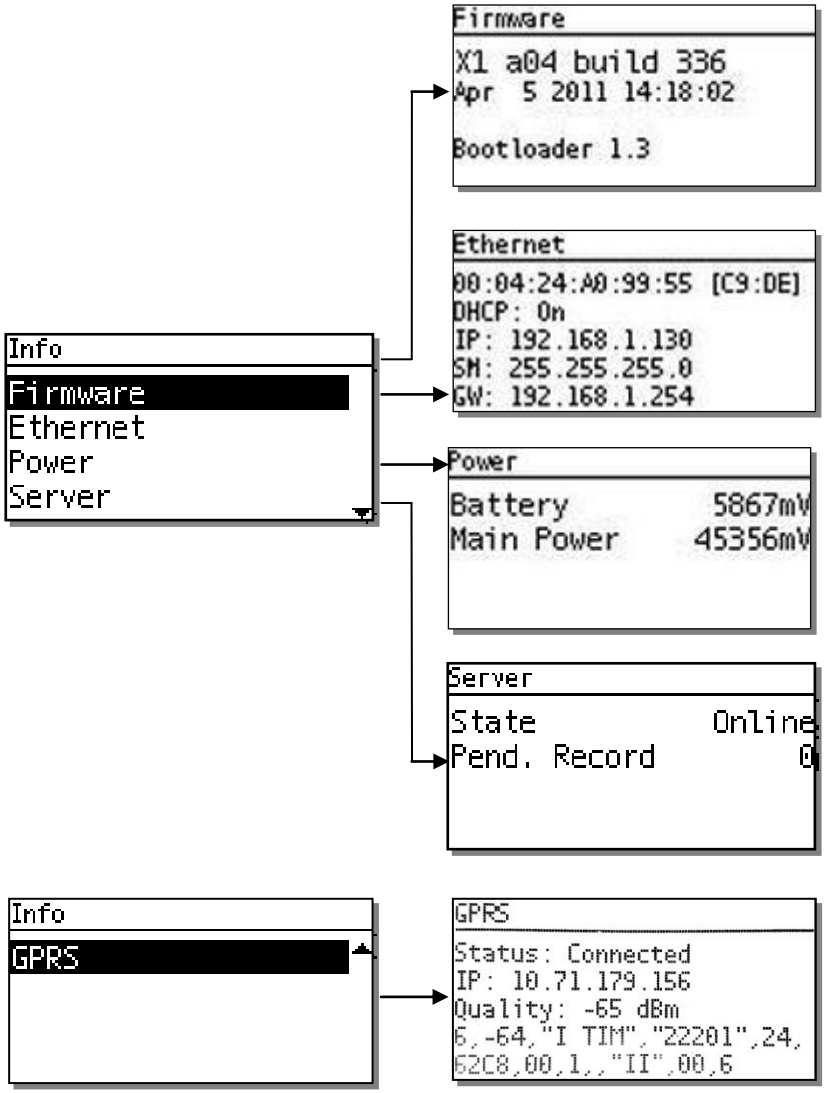
Il menu principale ("Configurazione") contiene 6 sezioni: Info, Ethernet, Display, Time & Date e (visibili solo spostando la selezione verso il basso fino alla pagina successiva) Biometrics e USB:





Usate i tasti freccia ▲▼ per selezionare una voce di menu e ↵ (Enter) per confermare.

Il menu "Info", a sua volta, contiene 5 sezioni: Firmware (versioni), Ethernet (attuali valori IP), Power (valori di tensione di alimentazione e batterie), Server (stato di comunicazione col server web e numero di transazioni non ancora inviate) e GPRS (visibile solo spostando la selezione verso il basso fino alla pagina successiva, e utile solo se il modem GPRS è presente e abilitato):



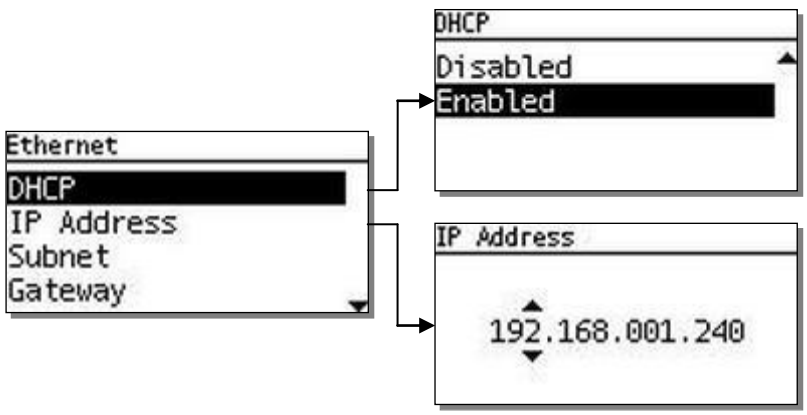
Il sottomenu “Info/Firmware” mostra l’attuale versione e la data di rilascio del firmware, e anche la versione di Bootloader.

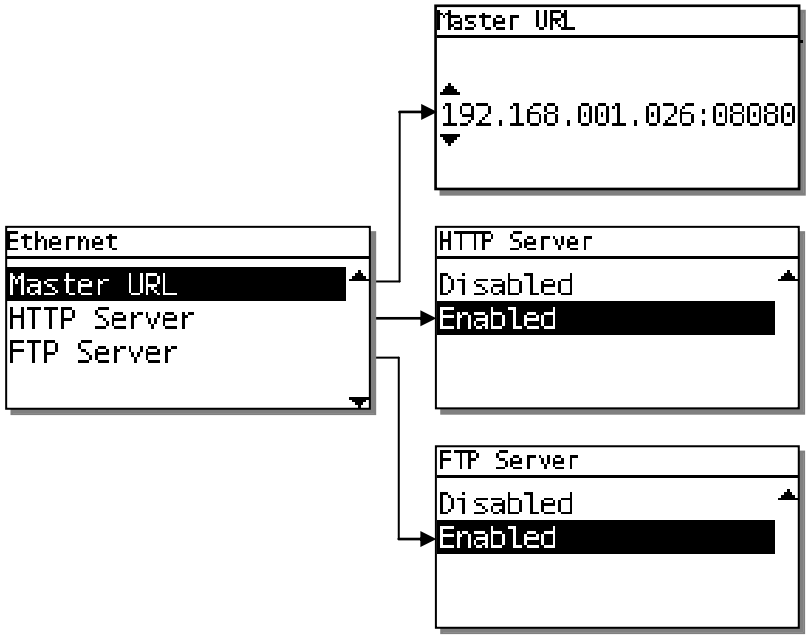
Il sottomenu “Info/Ethernet” mostra l’indirizzo MAC, lo stato DHCP e gli attuali indirizzo IP, subnet e gateway.

Il sottomenu “Info/Power” mostra i valori correnti sia della tensione di alimentazione che della batteria, e l’attuale stato della batteria.

Il sottomenu “Info/GPRS” mostra lo stato attuale del modem, l’indirizzo IP pubblico dinamico assegnato dal fornitore di servizi (solo quando il modem GPRS è connesso), l’attuale qualità del segnale e l’intera stringa con i dati relativi al segnale di rete cellulare, così come restituita dal modem GPRS.

Il menu “Ethernet” contiene 7 sezioni, una per ciascun parametro impostabile: modalità DHCP, indirizzo IP (usato solo se il DHCP è disabilitato, o nel caso in cui il server DHCP non risponda), subnet mask, indirizzo gateway e (visibile solo spostando la selezione verso il basso fino alla pagina successiva) Master URL, modalità HTTP server e modalità FTP server:

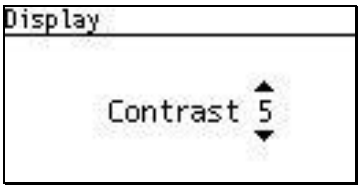




Usate ancora i tasti freccia ▲▼ per modificare una singola cifra (solo su X2, potete anche premere il tasto corrispondente sulla tastiera numerica), ↵ (Enter) per passare alla cifra successiva o per confermare quando vi trovate sull'ultima e **Clr** per tornare indietro di una posizione o abortire quando vi trovate sulla prima.

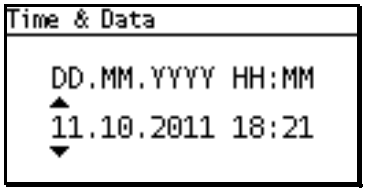
Nota: ciascun byte dell'indirizzo IP va sempre specificato in decimale su 3 cifre con eventuali zeri di riempimento a sinistra; i punti di separazione fra i byte sono fissi e vengono automaticamente saltati quando si procede all'impostazione della cifra successiva. Nel solo parametro MasterURL, il carattere ":" è usato per separare l'indirizzo IP dalla porta usata: anch'esso è fisso e viene saltato automaticamente, mentre il numero della porta (se specificata) va espresso in decimale su 5 cifre con eventuali zeri di riempimento a sinistra.

Per regolare la visibilità dello schermo selezionate la voce "Display" e quindi "Contrast" (valori da 0 a 9, il default è 5):



Usate ancora i tasti freccia ▲▼ per modificare il valore (solo su X2, potete anche premere il tasto corrispondente sulla tastiera numerica), ↵ (Enter) per confermare e **Clr** per abortire.

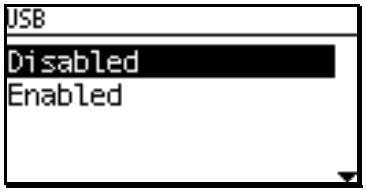
La voce "Time & Date" consente di impostare data e ora manualmente, utile nel caso in cui X1/X2 venga utilizzato come terminale *standalone* senza un collegamento Ethernet (il che rende impossibile l'impostazione dell'ora tramite il web server HTTP o tramite il caricamento di un file via FTP, come descritto al §4.1 a pag. 16):



Data e ora vanno inserite seguendo l'ordine GG.MM.AAAA HH:MM. Usate ancora i tasti freccia ▲▼ per modificare una singola cifra (solo su X2, potete anche premere il tasto corrispondente sulla tastiera numerica), ↵ (Enter) per passare alla cifra successiva o per confermare quando vi trovate sull'ultima e **Clr** per tornare indietro di una posizione o abortire quando vi trovate sulla prima. Nota: se viene inserito un valore non valido (ad esempio una data non esistente), il terminale non accetta la conferma finale, senza però mostrare nessun messaggio di errore: usate **Clr** per tornare indietro e correggere.

Per una descrizione dettagliata del menu "Biometrics" si veda il §11.1 a pag. 123.

La voce “USB”, infine, consente semplicemente di abilitare o disabilitare la gestione delle chiavette di memoria USB (solo su versioni di hardware **006** e successive, vedi §14 a pag. 148): in pratica permette di impostare il parametro **Enable** all’interno della sezione *[USB]* del file PARAMETERS.TXT (vedi §4.11 a pag. 59) anche nel caso in cui non sia possibile accedere al terminale via Ethernet neppure per la prima configurazione (e proprio per questo motivo si voglia appunto usare la funzionalità di scarico delle transazioni su chiavetta USB). Non si tratta comunque del menu di gestione delle chiavette USB vero e proprio (descritto al §14 a pag. 148), il quale comparirà solo in seguito all’inserimento della chiavetta una volta abilitata la gestione.

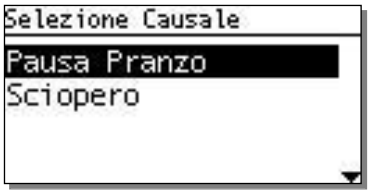


Usate i tasti freccia ▲▼ per selezionare un’opzione e ↵ (Enter) per confermare.

In ogni schermata potete premere il tasto **[<-]->** per abortire e tornare immediatamente alla schermata precedente, e infine uscire dal menu supervisore.

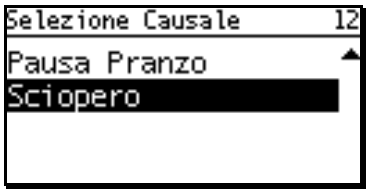
10.6 TRANSAZIONI CON CODICE CAUSALE

Se è stato precedentemente caricato un file di testo chiamato REASONS.TXT (vedi §4.4 alla pag. 19), o in alternativa un altro file chiamato AXREASON.TXT (vedi §5.10 alla pag. 79), dalla schermata di stand-by (attesa lettura carta), premendo il tasto “freccia giù” (▼) si attiva il menu “Selezione Causale”, che si può scorrere con i tasti ▲▼ (c’è un timeout di 10 secondi):



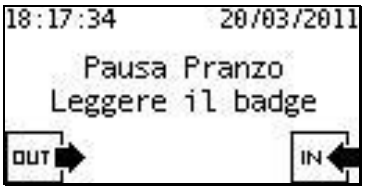
Nota: tenendo premuti i tasti ▲▼ è possible scorrere velocemente la lista in entrambi i sensi (modalità *auto scroll*).

Solo sui modelli X2 con tastiera numerica è anche possibile, una volta entrati nel menu “Selezione Causale”, premere un tasto numerico **1..9** per posizionarsi automaticamente sulla prima causale avente un codice che inizia con tale cifra, mentre la cifra inserita viene mostrata in alto a destra. Premendo ulteriori tasti numerici, in maniera analoga, ci si posiziona sulla prima causale il cui codice inizia con l’intera sequenza di cifre digitate, che viene di volta in volta aggiornata nella visualizzazione in alto a destra:



Se non viene trovata nessuna corrispondenza, l’intera sequenza di cifre inserita viene automaticamente cancellata e si può ripartire con un'altra chiave di ricerca inserendo di nuovo la prima cifra.

↵ (Enter) seleziona la causale evidenziata, quindi il terminale si pone in attesa della carta che sarà associata al codice causale (timeout di 10 secondi):



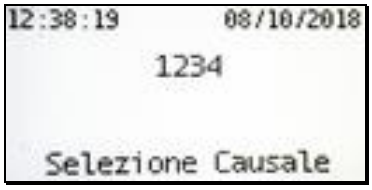
La carta utente può anche essere letta mentre la causale desiderata è evidenziata, evitando di dover usare il tasto ↵ (Enter).

Se è stato caricato anche un file chiamato FKEY.TXT (vedi §4.5 a pag. 20), è possibile effettuare una selezione diretta (scelta rapida) della causale, cioè evitare di passare attraverso il menu di selezione premendo semplicemente un tasto numerico (solo sui modelli X2 con tastiera numerica) per selezionare la causale associata, come descritto al §10.2 a pag. 108. Esattamente come con la selezione standard da menu, il terminale si pone quindi in attesa della carta che sarà associata al codice causale (timeout di 10 secondi).

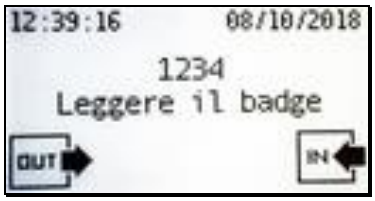
Se la transazione è accettata, compare la conferma della timbratura con verso di passaggio, codice della tessera e la causale selezionata:

Nel caso in cui si usi il file AXREASON.TXT, il controllo degli accessi sia attivato (vedi §5 a pag. 68), e siano stati caricati i file CARDS.TXT (§5.4 a pag. 71) e USERS.TXT (§5.9 a pag. 77), se la tipologia dell'utente relativo al codice tessera letto è compatibile con la causale selezionata, compare la conferma della timbratura con verso di passaggio, nome dell'utente contenuto nel file USERS.TXT e causale selezionata:

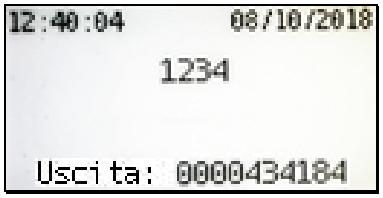
E' anche possibile abilitare la funzionalità di introduzione causali numeriche "libere" (ovvero non predefinite in REASONS.TXT né in AXREASON.TXT) mediante la digitazione manuale del codice causale prima di effettuare la lettura di tessera. A tale scopo è possibile impostare il par. **AllowTypeReason** nella sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 35) ad un valore diverso da '0', oppure associare un tasto di scelta rapida a tale funzionalità (vedi file FKEY.TXT al §4.5 a pag. 20). Quale che sia il metodo utilizzato per passare alla digitazione della causale numerica "libera", il codice causale inserito può essere lungo a piacimento (come per le causali predefinite, da un minimo di 1 ad un massimo di 8 cifre):



Una volta terminata la digitazione è possibile confermarlo premendo il tasto ↵ (Enter) e quindi leggere una tessera valida per effettuare la timbratura:

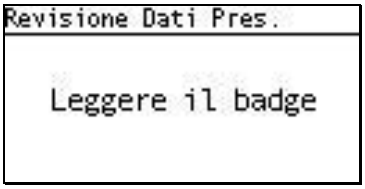


O anche saltare la conferma passando direttamente alla lettura della tessera:



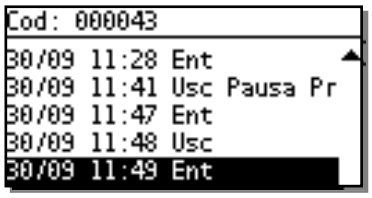
10.7 REVISIONE DATI DI PRESENZA

Dallo schermata di stand-by premete il tasto “freccia su” (▲): se non è stato caricato un file ENQUIRY.TXT (vedi nota (**) al §4.5 a pag. 20), viene mostrata la richiesta di lettura del badge per poter effettuare la revisione dei dati di presenza di un utente:



A questo punto è possibile solo leggere la carta di un utente (c’è un timeout di 10 secondi) oppure, se è consentito effettuare transazioni digitando i codici manualmente (parametro **AllowTypeCode**=1 all’interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT, vedi §4.11 a pag.31), è anche possibile digitare manualmente il codice di cui si vogliono visualizzare le transazioni precedentemente effettuate. Se si desidera disabilitare questa opzione, ad esempio per motivi di privacy, si può impostare il parametro **DisableTypeCodeReviewTA**=1, sempre all’interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT.

Scorrete poi le transazioni memorizzate nel file system sulla micro-SD del terminale, partendo dall’ultima transazione effettuata, con i tasti ▲▼:

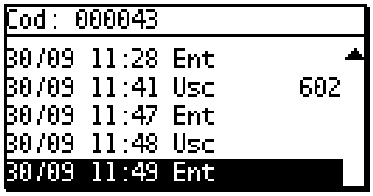


Nota 1: tenendo premuti i tasti ▲▼ è possible scorrere velocemente la lista in entrambi i sensi (modalità *auto scroll*).

Nota 2: è irrilevante se il file TRANSACTIONS.TXT sia stato cancellato oppure no, perché il terminale conserva sempre una copia delle transazioni effettuate nel file riservato **btransactions.loc**, vedi §7 a pag. 96).

Ogni riga corrisponde ad una singola transazione e riporta, nell’ordine: data e ora (nel formato “GG/MM HH:mm”), verso di passaggio su 3 caratteri (Ent/Usc) e i primi 8 caratteri della descrizione dell’eventuale causale associata alla timbratura, contenuta nel file REASONS.TXT (vedi §4.4 alla pag. 19) o, in alternativa, nel file AXREASON.TXT (vedi §5.10 alla pag. 79):

Se, successivamente alla registrazione di una transazione con causale, tale causale viene rimossa dal file che la definiva (REASONS.TXT o AXREASON.TXT), o l’intero file viene rimosso (si ricordi che caricare AXREASON.TXT equivale a cancellare un eventuale REASONS.TXT già presente: quest’ultimo infatti non verrà più considerato poiché meno prioritario), il terminale non riuscirà più a trovare la descrizione associata al codice causale registrato nel file TRANSACTIONS.TXT, pertanto sarà in grado solo di mostrarne il codice:



Premete **Clr** o attendete 10 secondi per tornare alla schermata di stand-by.

Nota 1: la funzionalità di revisione dati di presenza può essere disabilitata in qualunque momento impostando a 1 il parametro **DisableReviewTA** nella sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 31) oppure, analogamente, spuntando la checkbox “**Disable Attendance Review**” nella pagina “**Time & Attendance**” del web server HTTP. In tal caso, premendo il tasto “freccia su” (▲) dalla schermata di stand-by non succede nulla.

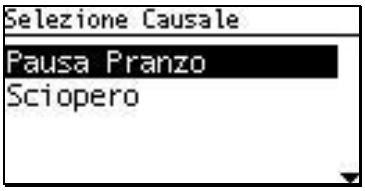
Nota 2: è possibile limitare il numero dei giorni precedenti al giorno corrente per i quali si possono visualizzare le transazioni effettuate, modificando il valore del parametro **ReviewDaysTA** nella sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 31) oppure, analogamente, impostando il campo di testo “**Attendance Review Days**” nella pagina “**Time &**

Attendance" del web server HTTP: il valore di default è 30. Ad esempio, impostando questo parametro al valore '1', verranno visualizzate solo le timbrature di oggi e di ieri, mentre con '0' verranno visualizzate solo le timbrature di oggi.

Nota 3: se è stato caricato un file ENQUIRY.TXT (normalmente non presente), selezionando l'opzione "**Revisione Dati Pres.**" dal menu utente, invece della richiesta di lettura badge come descritto in precedenza, compare un menu di selezione delle *enquiries* definite in tale file. Se lo si desidera, è comunque possibile aggiungere a tale menu anche l'opzione per il lancio della procedura standard di revisione dei dati di presenza: a tale scopo, è sufficiente aggiungere (nella posizione desiderata) un record al file ENQUIRY.TXT avente come identificativo il valore '99' (vedi §4.5 a pag. 20).

10.8 MENU "RIDOTTO" PER SELEZIONE CAUSALI / *ENQUIRIES* REMOTE

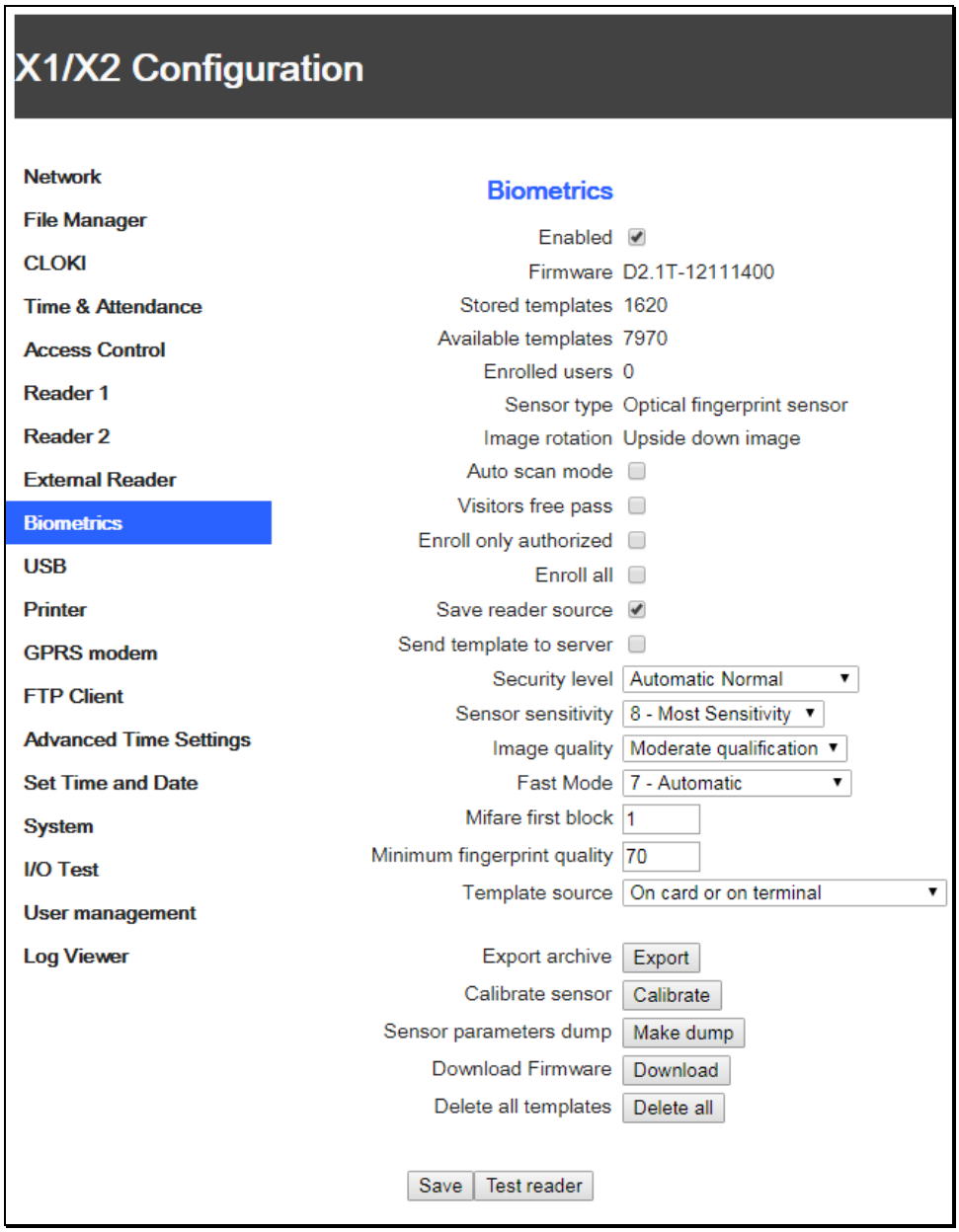
Se è stato precedentemente caricato un file di testo chiamato FKEY.TXT (vedi §4.5 alla pag. 20), dalla schermata di stand-by (attesa lettura carta), premendo il tasto ↵ (Enter) si attiva un menu "ridotto" contenente le descrizioni delle sole causali e/o delle *enquiries* remote (queste ultime disponibili solo se X1/X2 viene gestito dal programma Xatl@s) associate a tasti numerici per la selezione diretta (scelta rapida). Nonostante la scelta rapida sia disponibile solo sui modelli X2 con tastiera numerica, questo menu è accessibile anche sui modelli X1, ed il suo aspetto e funzionamento sono del tutto simili al menu standard di selezione della causale descritto al §10.6 a pag. 116: le voci si possono scorrere con i tasti ▲▼ e selezionare col tasto ↵ (Enter) (c'è un timeout di 10 secondi).



In questo caso, tuttavia, premendo un tasto numerico **1..9** si esce dal menu (non è possibile posizionarsi automaticamente sulla prima causale avente un codice che inizia con la cifra digitata).

11. IL MODULO BIOMETRICO ESTERNO FINGERBOX

X1/X2 può essere equipaggiato con un modulo biometrico esterno FingerBOX per la scansione di impronte digitali. Una volta inserito l'apposito connettore molex e attivata la gestione del FingerBOX da parte del terminale come descritto al §3.6 a pag. 11, è possibile ottenere informazioni sullo stato corrente del modulo biometrico e configurarne alcuni parametri caratteristici mediante la pagina "**Biometrics**" del web server HTTP del terminale (si noti che quando la gestione del FingerBOX è abilitata il caricamento di tale pagina richiede leggermente più tempo rispetto a tutte le altre in quanto include informazioni contenute solo all'interno del modulo biometrico e che devono pertanto essere nuovamente richieste ad ogni accesso alla pagina). E' comunque possibile editare gli stessi parametri direttamente all'interno della sezione *[Biometric]* del file PARAMETERS.TXT, vedi §4.11 a pag. 48. La pagina "**Biometrics**" del web server HTTP ha l'aspetto mostrato in figura:



Nell'ordine, compaiono:

- la *checkbox* per l'abilitazione della gestione del FingerBOX (corrispondente al parametro **Enabled**)
- la versione di firmware del modulo biometrico (da non confondere con la versione di firmware del terminale)

- il numero di *template* (cioè l'insieme dei dati binari generati da ogni singola scansione di impronta) già memorizzati all'interno della memoria del modulo biometrico
- il numero di *template* che è ancora possibile memorizzare (pari a 9590, chè è il limite massimo, meno il numero di *template* già memorizzati)
- il numero degli utenti biometrici già registrati (per ciascun utente è possibile scansionare fino a 2 impronte diverse, per ciascuna delle quali vengono memorizzati 2 *template*)
- il tipo di sensore di impronte collegato al modulo biometrico
- la *checkbox* per l'abilitazione della modalità "autoscan" (o "identificazione 1:N", corrispondente al parametro **FreeScan**): in questo caso il sensore di impronte rimane sempre in uno stato di attesa scansione, pertanto è possibile appoggiare direttamente il dito sul sensore in qualunque momento e procedere all'identificazione dell'utente sulla base del confronto del *template* appena scansionato con tutti quelli già presenti nel modulo. Per questo motivo si può anche parlare di modalità "solo dito".
  Nella modalità di default "solo verifica 1:1", invece, l'utente deve prima identificarsi mediante la lettura di una tessera o (ma solo sui modelli X2 con tastiera numerica, e se il parametro **AllowTypeCode** all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT è impostato a 1, vedi §4.11 a pag.31) tramite la digitazione manuale del codice tessera, e solo a questo punto il terminale pone il sensore in stato di attesa scansione e chiede all'utente di appoggiare il dito per poi effettuare la verifica di identità confrontando il *template* appena scansionato solo con quelli relativi al codice tessera introdotto.

  Nota: in modalità "autoscan" è comunque possibile procedere con la "verifica 1:1" inserendo il codice utente invece di appoggiare direttamente il dito sul sensore.
- la *checkbox* per l'esenzione degli utenti "visitatori" (per i quali non è prevista la registrazione nel sistema biometrico) dalla richiesta di verifica 1:1 (corrispondente al parametro **FreePass**, vedi ulteriori dettagli al §11.4 a pag. 135)
- la *checkbox* per consentire la registrazione di impronte solo agli utenti autorizzati mediante un apposito flag opzionale all'interno del file CARDS.TXT (corrispondente al parametro **EnrollAuth**, vedi ulteriori dettagli al §5.4 a pag. 71)
- la *checkbox* per consentire la registrazione di impronte per tutti i codici tessera inseriti (corrispondente al parametro **EnrollAll**). Normalmente invece, se è presente almeno un file fra CARDS.TXT e CARDRNGE.TXT, il codice inserito viene accettato solo se è fra quelli elencati in CARDS.TXT o si trova all'interno di un intervallo di codici elencato in CARDRNGE.TXT.
- la *checkbox* per abilitare/disabilitare l'uso del campo "provenienza lettura" nei record aggiunti ai file USERCODS.TXT e BIOUPDATE.TXT dopo ogni registrazione di impronta di un nuovo utente (corrispondente al parametro **SaveReaderSource**, vedi ulteriori dettagli al §11.1 a pag. 127)
- la *checkbox* per abilitare/disabilitare la trasmissione online HTTP immediata di tutti i nuovi record memorizzati nel file BIOUPDATE.TXT, che possono essere relativi a qualunque operazione effettuata all'interno del menu di gestione dell'archivio di impronte (corrispondente al parametro **SendTemplate**, vedi ulteriori dettagli al §4.11 a pag. 50).
- il menu a tendina per impostare il livello di sicurezza del modulo biometrico, corrispondente al parametro **SecurityLevel**. Tale parametro può assumere valori da 1 a 18 (default 16), con il seguente significato:

  1..15: livello fisso → 1: sicurezza minima .. 15:sicurezza massima

  16..18: livello variabile automaticamente in base al numero di *template* memorizzati → 16: normale, 17: sicuro, 18: più sicuro)

  Il livello di sicurezza specifica il FAR (*False Acceptance Ratio*, cioè rapporto di falsa accettazione): valori di sicurezza più alti corrispondono a valori FAR più bassi. Ad esempio un FAR di 1/100.000 significa che la probabilità di accettare impronte non autorizzate è pari a 1/100.000. Poiché FAR e FRR (*False Rejection Ratio*, cioè rapporto di falso rifiuto) sono inversamente proporzionali fra loro, l'FRR aumenta con maggiori livelli di sicurezza. In modalità "identificazione 1:N", il FAR aumenta: in tal caso, pertanto, raccomandiamo di impostare valori di sicurezza più alti (quindi un FAR più basso), soprattutto quando nel modulo sono memorizzate diverse centinaia di *template*. Quando si imposta un valore automatico (16..18), il livello di sicurezza viene regolato automaticamente per ottenere i seguenti valori FAR in base alla modalità di utilizzo e al numero di *template* memorizzati nel modulo:

Livello automatico	Verifica (1 :1)	Identificazione (1 :N)			
		1 ~ 9	10 ~ 99	100 ~ 999	1000 ~ 9999
16 (normale)	1/10,000	1/10,000	1/100,000	1/1,000,000	1/10,000,000
17 (sicuro)	1/100,000	1/100,000	1/1,000,000	1/10,000,000	1/100,000,000
18 (più sicuro)	1/1,000,000	1/1,000,000	1/10,000,000	1/100,000,000	1/100,000,000

- il menu a tendina per impostare la sensibilità di rilevamento del sensore, corrispondente al parametro **Sensitivity**. Tale parametro può assumere valori da 1 (sensibilità minima) a 8 (sensibilità massima - default). Con una sensibilità alta il modulo biometrico accetta più facilmente l'impronta immessa, mentre con una minore sensibilità l'immagine dell'impronta immessa sarà più stabile.
- il menu a tendina per impostare il livello di qualità dell'immagine affinché una scansione dell'impronta possa essere considerata, sia in fase di registrazione che in fase di riconoscimento (corrispondente al parametro **ImageQuality**). Tale parametro può assumere valori da 1 (accetta qualità minima) a 4 (richiede qualità massima), il valore di default è 2. Quando viene scansionata un'impronta, il modulo biometrico controlla se la qualità dell'immagine è adeguata per essere elaborata ulteriormente. Se è scarsa, il modulo biometrico invia un messaggio d'errore. Il parametro corrispondente specifica la severità di questo controllo di qualità.
- il menu a tendina per impostare la velocità di identificazione 1:N, corrispondente al parametro **FastMode**. Tale parametro può assumere valori da 1 a 7 (default 7), con il seguente significato:

  1..6: velocità fissa → 1: normale (più lenta) .. 6: velocità massima

  7: velocità variabile automaticamente in base al numero di *template* memorizzati nel modulo

  Quando in un modulo biometrico sono salvate diverse centinaia di *template*, il tempo necessario per l'identificazione 1:N (ricerca della corrispondenza fra le impronte) può risultare molto lungo: questo parametro può in tal caso essere utilizzato per ridurre il tempo di identificazione, a scapito di un leggero peggioramento del risultato di autenticazione. Il FAR non viene influenzato da questo parametro, ma l'FRR può risultare leggermente superiore rispetto alla modalità normale (valore "1"). Tipicamente, il valore "2" è 2-3 volte più veloce della modalità normale, mentre il valore "6" è 6-7 volte più veloce, sempre rispetto alla modalità normale. Con il valore di default "7", il valore effettivo viene regolato automaticamente in base al numero di *template* memorizzati nel modulo, secondo la tabella seguente:

***Template* registrati**	**FastMode**
1 ~ 99	1 (normale)
100 ~ 499	2
500 ~ 999	3
1000 ~ 1999	4
2000 ~ 3999	5
4000 ~ 9999	6

- la casella di testo per impostare il numero (in decimale) del blocco dati a partire dal quale è possibile memorizzare un *template* all'interno di una carta di prossimità Mifare R&W, corrispondente al parametro **MifareFirstBlock**. Questo parametro ha effetto solo se si intende salvare i *template* di ciascun utente direttamente all'interno di una carta Mifare personale (vedi §11.2 a pag. 132): i dati biometrici vengono in tal caso scritti su tutti i blocchi dati adiacenti disponibili a partire da quello specificato da questo parametro (default 1, cioè il primo blocco disponibile dopo il blocco 0 a sola lettura che contiene anche il codice univoco UID).
- la casella di testo per impostare il valore minimo del punteggio (*score*) relativo ai *template* affinché essi possano essere memorizzati in fase di registrazione delle impronte, corrispondente al parametro **MinimumQuality**. Tale parametro può assumere valori da 0 a 100, ma si raccomanda di usare valori maggiori o uguali a 70 (default). Nota: lo *score* non dipende dalla qualità dell'immagine dell'impronta, bensì dal solo contenuto informativo rilevante ai fini del riconoscimento biometrico (*minuzie*) e dalla corrispondenza fra i dati relativi alle due scansioni effettuate in fase di registrazione delle impronte.

- il menu a tendina per specificare dove debbano essere cercati i *template* registrati al momento di effettuare l'autenticazione biometrica, corrispondente al parametro **TemplateSource**. Tale parametro può assumere valori da 0 a 3 (default 2), con il seguente significato:

  0 → ricerca solo all'interno della carta appena letta. Questa opzione può essere usata solo se si utilizzano carte di prossimità Mifare R&W su ciascuna delle quali sono stati in precedenza memorizzati i *template* del possessore della carta (vedi §11.2 a pag. 132), e solo in modalità "carta + dito": una volta salvata questa impostazione, l'eventuale spunta sulla *checkbox* per l'abilitazione della modalità "autoscan" viene automaticamente rimossa (cioè il parametro **FreeScan** viene reimpostato a 0). Usando un qualunque altro tipo di carta o la digitazione manuale del codice si ottiene sempre il messaggio di errore "**Tessera non valida**".

  1 → ricerca solo sul terminale (file USERCODS e memoria interna del modulo FingerBOX). Questa opzione va usata solo se non si vogliono mai utilizzare per la verifica biometrica 1:1 i *template* eventualmente memorizzati su carte di prossimità Mifare R&W.

  2 (default) → ricerca prima all'interno della carta appena letta, poi (solo se non trova nulla) sul terminale

  3 → ricerca prima sul terminale, poi (solo se non trova nulla) all'interno della carta appena letta

Vi sono poi alcuni pulsanti che non sono relativi alla configurazione del funzionamento del modulo biometrico, ma consentono invece di effettuare alcune operazioni importanti, presenti come opzioni anche nella sezione "Biometrics" del menu supervisore accessibile dalla tastiera del terminale (vedi paragrafo successivo):

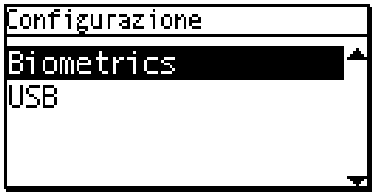
- "Export Archive": esportazione dell'intero archivio biometrico, cioè dei dati relativi a tutte le impronte attualmente presenti nel modulo (vedi anche §11.1 a pag. 130)
- "Delete all templates": cancellazione dell'intero archivio biometrico, e di tutti i file contenenti dati ad esso relativi presenti nel file system del terminale (vedi anche §11.1 a pag. 130)
- "Calibrate sensor": ricalibrazione del sensore, da effettuare nel caso si verifichino peggioramenti nelle prestazioni del sensore (ad esempio diversi utenti che in precedenza venivano riconosciuti senza problemi ad un certo punto non lo sono più, vedi anche §11.1 a pag. 131)

Infine è disponibile un pulsante utilizzabile a solo scopo di debug:

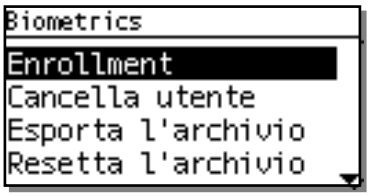
- "Sensor parameters dump": scrittura del valore di tutti i parametri del modulo biometrico all'interno del file LOG.TXT

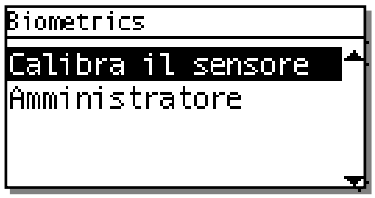
11.1 MENU DI GESTIONE DELL'ARCHIVIO DELLE IMPRONTE

L'archivio delle impronte contenuto nel modulo biometrico può essere gestito direttamente dalla console di X1/X2 mediante la sezione "Biometrics" del menu supervisore descritto al §10.5 a pag. 113.



Tale sezione compare fra le opzioni del menu anche se la gestione del FingerBOX non è stata attivata, ma in tal caso selezionandola appare solo il messaggio di errore "Biometria disabilitata". Se invece la gestione è attiva vengono visualizzate le 6 voci disponibili: "Enrollment", "Cancella utente", "Esporta l'archivio", "Resetta l'archivio" e (visibili solo spostando la selezione verso il basso fino alla pagina successiva) "Calibra il sensore" e "Amministratore":





Nota 1: solo nel caso in cui almeno uno dei parametri **CardDecode** all'interno delle sezioni *[Reader1]*, *[Reader2]* e *[ExtReader]* del file PARAMETERS.TXT (vedi §4.11 a pag. 42) sia impostato ai valori '30' / '32' / '41' / '42' / '43' (quelli relativi ad un lettore RFID2/3 seriale TTL 13,56MHz), la sezione "Biometrics" contiene una ulteriore voce "Canc tessera Mifare": vedi §11.2 a pag. 132 per ulteriori dettagli.

Nota 2: solo nel caso in cui si entri nel menu supervisore mediante identificazione biometrica di un utente con attributi di "amministratore solo biometrico", la voce "Amministratore" non compare fra le opzioni disponibili (vedi "AMMINISTRATORE" a pag. 131).

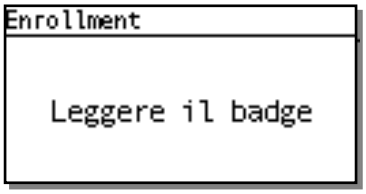
Nota 3: solo in caso di sensore di impronte di tipo ottico, la voce "Calibrate" non compare fra le opzioni disponibili in quanto non necessaria.

Usate i tasti freccia ▲▼ per selezionare una voce di menu e ↵ (Enter) per confermare (timeout: 30 secondi):

Vediamo ora ciascuna voce in dettaglio:

- **ENROLLMENT**

  E' la voce usata per la registrazione delle impronte. Come prima cosa X1/X2 chiede di leggere il badge dell'utente che deve registrare le sue impronte (timeout: 10 secondi):



  In alternativa, è sempre possibile inserire manualmente il codice utente:

  1) solo su X2, è possibile digitare il codice usando i tasti numerici (timeout: 10 secondi). **Nota**: è possibile inserire un numero di cifre minore o uguale al valore del parametro **CardCodeLength** all'interno della sezione *[Reader1]* del file PARAMETERS.TXT. Se il numero di cifre è minore, il codice verrà completato con riempimento di zeri a sinistra.

  2) sia su X1 che su X2, è comunque possibile inserire il codice usando i tasti ▲▼: alla prima pressione compare un campo numerico con un numero di cifre pari al valore del parametro **CardCodeLength** appena citato, inizialmente impostate tutte a '0' (timeout: 10 secondi): a questo punto potete nuovamente usare i tasti ▲▼ per modificare ciascuna singola cifra (solo su X2, potete anche premere il tasto corrispondente sulla tastiera numerica), **Clr** per tornare indietro di una posizione o abortire quando vi trovate sulla prima e ↵ (Enter) per passare alla cifra successiva o per confermare quando vi trovate sull'ultima.

  Solo in caso di lettura da badge, viene comunque chiesta conferma del codice personale estratto (timeout: 10 secondi):

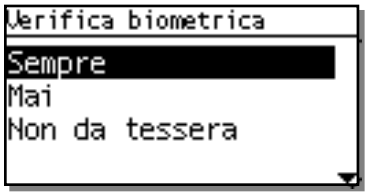
  Anche in questo caso, se volete, potete usare i tasti ▲▼ per modificare ciascuna singola cifra, **Clr** per tornare indietro di una posizione o abortire quando vi trovate sulla prima e ↵ (Enter) per passare alla cifra successiva o per confermare quando vi trovate sull'ultima (come quando si è appena effettuata la lettura).

  In tutti i casi sopra citati è anche possibile abortire direttamente mediante i tasti **.F** o **[<-]->**.

  Una volta confermato il codice, X1/X2 controlla se sia stato caricato almeno un file fra CARDS.TXT e CARDRNGE.TXT (vedi §5.1 a pag. 68), e se li trova verifica che il codice inserito sia fra quelli elencati in CARDS.TXT (in questo caso anche il flag **R** viene controllato, vedi dettagli al §5.4 a pag. 71) o si trovi all'interno di un intervallo di codici elencato in CARDRNGE.TXT. Questa verifica viene effettuata sempre in caso di presenza di tali file, a prescindere dal fatto che il controllo accessi sia attivato o meno, e dà esito positivo anche se la tessera o l'intervallo sono presenti ma definiti come "disabilitati". Inoltre, se il parametro **EnrollAuth**=1 nella sezione *[Biometric]* del file PARAMETERS.TXT (vedi §4.11 a pag. 49) viene anche controllato il flag **B** (biometrico)nel file CARDS.TXT, che in questo caso deve avere il formato "esteso" (cioè con tutti i record di 71 caratteri invece che 69 come nella versione standard): solo se tale flag è impostato a '1' il codice tessera è autorizzato alla registrazione di impronte. In caso di verifica negativa, non è possibile procedere con la registrazione delle impronte per il codice inserito, e compare il messaggio di errore "**Operazione fallita – Tessera non valida**". Tutti i controlli appena descritti possono comunque essere saltati impostando il parametro **EnrollAll**=1 nella sezione *[Biometric]* del file PARAMETERS.TXT (vedi §4.11 a pag. 49): in questo caso verranno sempre accettati tutti i codici inseriti.

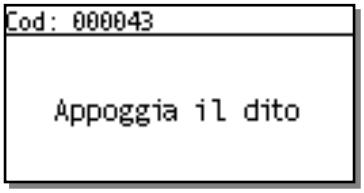
Se il codice viene accettato e non risulta essere già presente nessuna impronta associata al codice inserito, appare la schermata per la selezione della modalità di richiesta della verifica biometrica 1:1 per quel codice (timeout: 30 secondi; vedi anche al §11.4 a pag. 135 le ulteriori modalità di esenzione dalla verifica biometrica). Se invece è già presente almeno un'impronta associata a quel codice, ciò significa che tale scelta deve essere già stata fatta in precedenza, per cui si passa direttamente alla scansione dell'impronta (vedi più avanti).

Nota: solo nel caso in cui il parametro **CardDecode** all'interno della sezione *[Reader1]* / *[Reader2]* / *[ExtReader]* del file PARAMETERS.TXT (vedi §4.11 a pag. 42) relativa al lettore usato per inserire il codice sia impostato ai valori '30' / '32' / '41' / '42' / '43' (quelli relativi ad un lettore RFID2/3 seriale TTL 13,56MHz), prima della schermata "Verifica biometrica" descritta nel seguito può comparirne un'altra ("Salvataggio template") relativa al salvataggio delle impronte su carte Mifare, vedi §11.2 a pag. 132 per ulteriori dettagli.

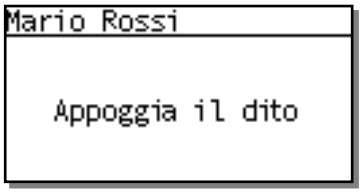


Selezionando "Sempre", ogni volta che nello stato di attesa transazioni si leggerà questa tessera o si digiterà manualmente questo codice (solo su X2 e se il parametro **AllowTypeCode**=1 all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT, vedi §4.11 a pag.31) verrà richiesto di appoggiare il dito sul sensore per procedere alla verifica biometrica (con le eccezioni descritte al §11.4 a pag. 135), altrimenti non sarà possibile accettare la transazione. Selezionando "Mai", invece, la verifica biometrica non verrà mai richiesta in seguito alla lettura / digitazione di questo codice tessera: in pratica per questo utente si consentono le modalità di timbratura "solo tessera" e "solo digitazione manuale". Infine, selezionando "Non da tessera", la verifica biometrica non verrà richiesta, ma solo in seguito alla lettura di questa tessera, mentre non sarà comunque consentita la digitazione manuale di questo codice (che darebbe luogo al messaggio di errore "**Non autorizzata**"): in pratica per questo utente si consente unicamente la modalità di timbratura "solo tessera". Indipendentemente dalla scelta effettuata, l'eventuale modalità "solo dito" si può abilitare (ma solo per tutti gli utenti) come spiegato al §11 a pag. 121.

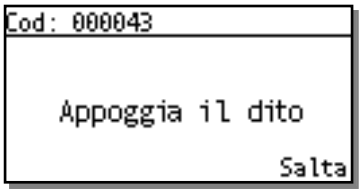
Il passo successivo è quello relativo alla scansione dell'impronta: X1/X2 mostra in alto a sinistra il codice inserito, oppure il nome dell'utente nel caso sia possibile risalirvi (cioè solo se sono stati caricati entrambi i file CARDS.TXT e USERS.TXT, a prescindere dal fatto che il controllo accessi sia attivato o meno), e chiede di appoggiare il dito sul sensore di impronte digitali (timeout: 10 secondi):



oppure



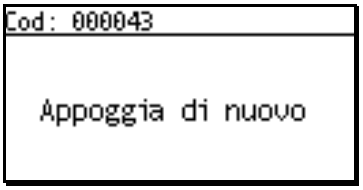
Nota: solo nel caso in cui abbiate appena selezionato la voce "Mai" oppure "Non da tessera" nella precedente schermata "Verifica biometrica", e se avete intenzione di non usare neppure la modalità "solo dito" per questo utente, è possibile saltare la fase di registrazione dell'impronta. La schermata ha infatti un'aspetto leggermente diverso, con l'opzione "Salta" visualizzata in basso a destra, di fianco al tasto ↵ (Enter):



Premendo effettivamente il tasto ↵ (Enter), l'attesa per la scansione viene interrotta, e viene semplicemente chiesta conferma (timeout: 20 secondi):

Premete ancora ↵ (Enter) (Si) per confermare (in questo caso verrà mostrato il messaggio "Operazione terminata") e **[<-]->** (No) per passare ad una nuova richiesta di scansione (schermata "Inserire nuovo dito?", vedi sotto).

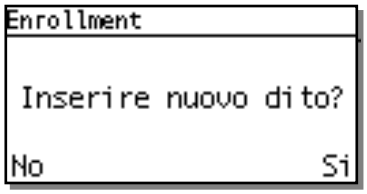
Tornando alla scansione dell'impronta, all'apparire della richiesta "Appoggia il dito" occorre: 1) posizionare il dito in una posizione in cui sia possibile sentire il bordo inferiore del FingerBOX in corrispondenza della seconda falange; 2) abbassare l'estremità del dito fino a farla aderire ad una superficie più ampia possibile del sensore (vedi figura); 3) tenere il dito fermo fino a quando sullo schermo non comparirà il messaggio seguente:



A questo punto occorre sollevare il dito dal sensore e ripetere i passi 1..3, fino a quando non si otterrà una richiesta di conferma di questo tipo (timeout: 20 secondi):

Quello che viene mostrato è un punteggio (*score*) relativo al solo contenuto informativo dell'impronta rilevante ai fini del riconoscimento biometrico (*minuzie*) e alla corrispondenza fra i dati relativi alle due scansioni effettuate. Il valore massimo è 100, ma vengono accettati anche valori più bassi, purché maggiori o uguali al valore del parametro **MinimumQuality** nella sezione *[Biometric]* del file PARAMETERS.TXT, vedi §4.11 a pag. 50 (default 70; cercate comunque di trovare il dito e la posizione che producano i migliori risultati). Premete ↵ (Enter) (Si) per confermare e **[<-]->** (No) per scartare le scansioni effettuate.

In entrambi i casi vi verrà proposto di ripetere la scansione per un nuovo dito (timeout:15 secondi), con identiche modalità:



Anche qui premete ↵ (Enter) (Si) per procedere e **[<-]->** (No) per uscire. Tenete presente che il modulo biometrico registra sempre entrambi i *template* relativi alle due scansioni effettuate per ciascuna impronta. In questo modo si dimezza il numero massimo di impronte registrabili, ma l'autenticazione viene migliorata, poiché in questo modo uno dei due *template* registrati potrà successivamente essere aggiornato in maniera automatica per riflettere i cambiamenti dinamici della pelle del dito dell'utente: ogni volta che ad un utente verrà chiesto di effettuare una verifica biometrica 1:1 durante una transazione, il modulo deciderà se il *template* esistente debba essere sostituito con quello nuovo appena ottenuto oppure no. ***Nota***: per applicazioni in rete con un server centrale che gestisce la distribuzione delle impronte, occorre prestare particolare attenzione, poiché in tal caso i cambiamenti automatici di un *template* possono causare problemi di sincronizzazione con il server.

Poiché ad ogni singolo utente è possibile associare un massimo di 4 *template*, questo significa che si possono salvare non più di 2 impronte ciascuno: se si cerca di effettuare la registrazione di un nuovo dito dopo avere già raggiunto il limite dei *template* registrati, compare il messaggio di errore "**Operazione Fallita – Limite impronte**".

In seguito ad ogni registrazione di impronta, il terminale salva i dati biometrici nella memoria interna del modulo FingerBOX, che può contenere fino a 9590 *template*, ciascuno identificato mediante un indice numerico a sole 4 cifre (quindi compreso fra 0001 e 9999) che definiremo *shortcode*. Questo indice viene generato in maniera sequenziale a partire da 0001, e incrementato ad ogni nuovo codice tessera per cui venga registrata una impronta, a meno che non siano state precedentemente cancellate le impronte già registrate per un altro codice tessera (vedi voce "CANCELLA UTENTE" a pag. 130), rendendo quindi riutilizzabile lo stesso *shortcode* ormai non più associato a quel codice tessera. L'associazione fra ciascun codice tessera ed il relativo *shortcode* è necessaria in verifica 1:1 per la ricerca dei *template* dell'utente (identificatosi tramite il codice tessera) all'interno della memoria del modulo per procedere al confronto con l'impronta da scansionare, ed in identificazione 1:N per risalire al codice tessera dell'utente (da registrare nel record della transazione) a partire dall'indice del *template* che è risultato corrispondere all'impronta appena scansionata. Questa associazione ed il meccanismo di generazione degli *shortcode* (e riutilizzo di quelli inutilizzati) vengono gestiti mediante il file di testo **USERCODS.TXT**. Questo file viene automaticamente

creato nella *root* del terminale in seguito al primo *enrollment*, e ad ogni nuovo codice tessera per cui venga registrata un'impronta(*) viene automaticamente aggiunto un record, o modificato un record già presente ma invalidato in seguito alla cancellazione di un utente.

I record di **USERCODS.TXT** hanno lunghezza fissa ed il seguente formato:

CCCCCCCCCCCCCCCC_SSSS_AAAAMMGG_N_T_A_R_M<CR><LF>

CCCCCCCCCCCCCCCC è il codice tessera su 16 cifre con eventuale riempimento di zeri a sinistra. Se il record viene invalidato in seguito alla cancellazione di un utente, le prime 10 cifre di questo campo vengono sovrascritte con altrettanti caratteri '$'=chr(36)

SSSS è lo *shortcode* utilizzato dal modulo biometrico per la registrazione delle impronte relative al codice tessera sopra menzionato. Questo campo non viene mai sovrascritto, anche se il record viene invalidato in seguito alla cancellazione di un utente: in questa maniera sarà possibile riutilizzare lo stesso *shortcode* alla successiva registrazione di un nuovo utente

AAAAMMGG è la data di creazione del record, o di riutilizzo di un record già esistente (ma precedentemente invalidato per cancellazione dell'utente) per un nuovo codice tessera

N è un flag ("NoFinger") che indica se e in quali casi l'utente sia esentato dalla richiesta di verifica biometrica, il cui valore è determinato dalla scelta effettuata nella schermata "Verifica biometrica" proposta all'utente prima di passare alla registrazione della sua prima impronta (vedi sopra). In seguito è possibile cambiare questa impostazione solo modificando direttamente il file USERCODS.TXT (attenzione: in questo caso leggete la nota a pag. 129):

0: assieme al codice tessera viene sempre chiesto anche il dito

1: utente esentato dalla verifica biometrica sia su lettura che su digitazione manuale del codice tessera

2: utente esentato dalla verifica biometrica solo su lettura della tessera (in seguito a digitazione manuale del codice viene chiesto anche il dito)

T è un identificatore del tipo di modulo biometrico utilizzato (attualmente fisso a '0', cioè modulo Suprema)

A è un flag che indica se questo utente abbia i diritti di amministratore (generale o solo biometrico, per i dettagli vedi voce "AMMINISTRATORE" a pag. 131). In seguito alla prima registrazione di impronta di un nuovo utente, questo campo è sempre fisso a '0' (utente normale). In seguito è possibile cambiare questa impostazione, sia mediante la voce di menu "AMMINISTRATORE" descritta a pag. 130 che modificando direttamente il file USERCODS.TXT (attenzione: in questo caso leggete la nota a pag. 129):

0: utente normale

1: amministratore di tipo "generale"

2: amministratore di tipo "solo biometrico"

R è un identificatore del lettore da cui proviene la lettura di tessera effettuata durante la registrazione di impronte in questione(*). Questo campo viene controllato durante ogni transazione effettuata in modalità "verifica 1:1": se il lettore usato per inserire il codice tessera non corrisponde con questo campo, la transazione non viene accettata, e viene mostrato il messaggio di errore "**Utente non trovato**". Inoltre, se il controllo accessi è abilitato, questo campo viene confrontato anche con l'analogo campo **R** all'interno dei record del file CARDS.TXT (vedi §5.4 a pag. 71 per ulteriori dettagli), e questo avviene anche per le transazioni effettuate in modalità "solo dito" (identificazione 1:N): in caso di mancata corrispondenza, la transazione non viene accettata e viene mostrato il messaggio di errore "**Tessera non valida**".

0: provenienza della lettura indifferente. Valore usato per le registrazioni di impronta effettuate su: 1) versioni di firmware precedenti alla a07_build863; 2) versioni di firmware a11_build250 e successive, ma solo se il parametro **SaveReaderSource**=0 nella sezione *[Biometric]* del file PARAMETERS.TXT, vedi §4.11 a pag. 50 (il default è 1, ovvero la provenienza delle letture viene considerata)

1: tessera letta sul lettore primario (READER1) o codice inserito manualmente

3: tessera letta sul lettore esterno su morsettiera a vite (EXTERNAL READER) o su eventuali lettori aggiuntivi collegati su schede di espansione 914 NeoMAX opzionali.

M è un flag relativo alla registrazione di impronta sotto minaccia (attualmente non gestito e quindi fisso a '0')

<CR><LF> sono 2 caratteri ASCII terminatori sempre presenti in coda ad ogni record, compreso l'ultimo (ne consegue che il file termina sempre con una linea vuota)

(*) ***Nota***: l'utente è ora strettamente legato non solo al codice tessera, ma anche al lettore utilizzato durante l'*enrollment*. Se si effettua un nuovo *enrollment* usando lo stesso codice tessera usato per una registrazione precedente ma effettuando la lettura su un lettore diverso, le impronte saranno considerate a tutti gli effetti quelle di un nuovo utente, e genereranno un nuovo record con diverso *shortcode* in USERCODS.TXT.

In seguito ad ogni registrazione di impronta, oltre ad aggiornare il file USERCODS.TXT, X1/X2 aggiorna anche il file di testo **BIOUPDATE.TXT**, il quale però non si trova nella *root* del terminale, bensì nella cartella **\BIOEXP** (normalmente non presente ma creata, assieme alla cartella **\BIOIMP** , subito dopo avere abilitato la gestione del modulo biometrico FingerBOX). Questo file viene automaticamente creato in seguito alla prima registrazione di impronta, e tiene traccia di tutte le operazioni effettuate all'interno del menu di gestione dell'archivio di impronte: registrazione di ogni nuovo dito, cancellazione di un utente, cambiamento degli attributi di un utente (esenzione dalla verifica biometrica, impostazione dei diritti di amministratore). Inoltre, contiene i *template* di tutti gli utenti registrati a partire dalla creazione del file. L'unica informazione non contenuta in questo file è quella relativa agli *shortcode* associati a ciascun codice tessera, che risiede solamente nel file USERCODS.TXT. Lo scopo di BIOUPDATE.TXT è consentire la sincronizzazione degli archivi di impronte contenuti nei moduli biometrici FingerBOX di altri terminali X1/X2 "slave" facenti parte dello stesso impianto, mediante il semplice trasferimento via FTP (o via chiavetta di memoria USB, vedi §14.5 e §14.6 a pag. 150) di questo file dal terminale "master" agli "slave" (e conseguente importazione automatica dei dati, vedi §11.3 a pag. 134), senza bisogno di ripetere tutte le singole operazioni in locale su ciascuno di essi. Una volta effettuata la sincronizzazione, BIOUPDATE.TXT può anche essere anche cancellato: verrà ricreato alla prima operazione effettuata e conterrà la "storia" dell'archivio a partire dall'ultima sincronizzazione effettuata. In questa maniera, la successiva sincronizzazione rappresenterà in effetti soltanto un aggiornamento, senza necessariamente dovere ricaricare su ciascuno "slave" l'intero archivio di impronte attualmente contenuto nel modulo biometrico del "master". A differenza di quanto accade per il file USERCODS.TXT, i record di BIOUPDATE.TXT, una volta generati, non vengono più modificati, qualunque operazione si faccia: semplicemente, ogni nuova operazione genera un nuovo record.

Nota: a partire dalla versione di firmware a11_build372, per default ogni nuovo record memorizzato nel file BIOUPDATE.TXT viene anche immediatamente trasmesso online via HTTP, assumendo che la gestione del protocollo sia abilitata, e che il parametro **MasterUrl** sia stato impostato (vedi §12 a pag. 137 per ulteriori dettagli): questo consente una gestione centralizzata del database di un impianto biometrico. Se non è necessaria, tuttavia, questa funzione può essere disabilitata impostando il par. **SendTemplate**=0 nella sezione *[Biometric]* del file PARAMETER.TXT (vedi 4.11 a pag. 50).

I record di **BIOUPDATE.TXT** possono essere di due tipi: il primo tipo (tipo A) è relativo ai record generati in seguito a ciascuna registrazione di impronta, che hanno lunghezza variabile ed il seguente formato:

CCCCCCCCCCCCCCCC_AAAAMMGG_nn_DDDD_ttt…**ttt_N_T_A_R_M_B_ff_E<CR><LF>**

CCCCCCCCCCCCCCCC è il codice tessera su 16 cifre con eventuale riempimento di zeri a sinistra

AAAAMMGG è la data di creazione del record (cioè la data di registrazione dell'impronta)

nn è il numero dei template contenuti nel record (max 04). Ciascun record di questo tipo contiene sempre tutti i *template* di un certo utente contenuti nell'archivio di impronte al momento dell'ultima registrazione effettuata da quell'utente, quindi se si effettua la registrazione di 2 impronte diverse dello stesso utente vengono comunque generati 2 record in BIOUPDATE.TXT: il primo contenente i 2 *template* relativi alla prima impronta, il secondo contenente tutti e 4 i *template* relativi ad entrambe le impronte

DDDD è la dimensione in byte di ciascun *template* contenuto nel record: sono supportati i formati a 384 byte (default) e a 256 byte (solo se importati da terminali 962 SuperTRAX opportunamente configurati per generare *template* in questo formato, vedi §11.3 a pag. 134)

ttt…**ttt** è la rappresentazione in ASCII-HEX del contenuto dei *template*: ogni coppia di caratteri rappresenta un byte in notazione esadecimale. La lunghezza di questo campo è quindi pari a [(nn x DDDD) x 2] caratteri, che equivalgono a (nn x DDDD) byte binari che sono stati memorizzati nella memoria interna del modulo biometrico

N è un flag che indica se e in quali casi l'utente sia esentato dalla richiesta di verifica biometrica: si tratta dello stesso campo presente anche nei record del file USERCODS.TXT (vedi sopra)

T è un identificatore del tipo di modulo biometrico utilizzato (attualmente fisso a '0', cioè modulo Suprema): si tratta dello stesso campo presente anche nei record del file USERCODS.TXT (vedi sopra)

A è un flag che indica se questo utente abbia i diritti di amministratore: si tratta dello stesso campo presente anche nei record del file USERCODS.TXT (vedi sopra).

R è un identificatore del lettore da cui proviene la lettura di tessera effettuata durante l'operazione in questione: si tratta dello stesso campo presente anche nei record del file USERCODS.TXT (vedi sopra). Nota: questo campo non viene comunque usato per la validazione delle transazioni, come invece accade per quello in USERCODS.TXT. E' però necessario in fase di importazione di impronte, proprio per la corretta impostazione del campo **R** nel record che viene creato all'interno di USERCODS.TXT.

M è un flag relativo alla registrazione di impronta sotto minaccia (attualmente non gestito e quindi fisso a '0'): si tratta dello stesso campo presente anche nei record del file USERCODS.TXT (vedi sopra)

B è un flag che identifica la categoria di dato biometrico registrato (attualmente fisso a '0', cioè impronta digitale): questo campo non è presente nei record del file USERCODS.TXT (vedi sopra)

ff è un identificatore del dito (fra i 10 disponibili sulle due mani: '01'..'10') a cui si riferiscono i *template* (attualmente fisso a '00', ovvero "non specificato"): questo campo non è presente nei record del file USERCODS.TXT (vedi sopra)

E è un flag che indica se i dati dei *template* sono specificati in formato standard ('0') o crittografato ('1'): questo campo non è presente nei record del file USERCODS.TXT (vedi sopra)

<CR><LF> sono 2 caratteri ASCII terminatori sempre presenti in coda ad ogni record, compreso l'ultimo (ne consegue che il file termina sempre con una linea vuota)

Il secondo tipo di record di **BIOUPDATE.TXT** (tipo B) è relativo ai record generati in seguito a tutte le altre operazioni effettuate all'interno del menu di gestione dell'archivio biometrico: registrazione di un utente senza *template* (in quanto esentato dalla verifica biometrica), cancellazione di un utente, impostazione dei diritti di amministratore. In questo caso il record ha lunghezza fissa ed il seguente formato:

CCCCCCCCCCCCCCCC_AAAAMMGG_nn_0000_N_T_A_R_M_B_ff_E<CR><LF>

Rispetto al caso precedente, notiamo che il campo **DDDD** è sempre fisso a "0000", e manca del tutto il campo **ttt**…**ttt** contenente i dati relativi ai *template* dell'utente. Inoltre:

nn può assumere solo i seguenti valori:

00: in caso di cancellazione dell'utente

99: in caso di semplice impostazione degli attributi dell'utente

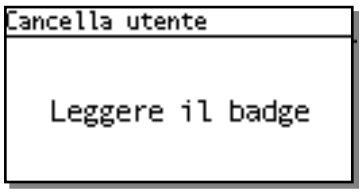
Tutti gli altri campi rimangono immutati. ***Nota***: il campo **E** in realtà non è rilevante (poiché questo secondo tipo di record non ha mai contenuti criptati), ma rispecchia semplicemente l'attuale valore del parametro **CryptoEnabled** nella sezione *[System]* del file PARAMETERS.TXT (vedi §4.11 a pag. 53).

Nota: le eventuali modifiche degli attributi di un utente effettuate modificando direttamente il file USERCODS.TXT non comportano la scrittura automatica di alcun record in BIOUPDATE.TXT, non essendo prodotte da un'operazione all'interno del menu di gestione dell'archivio. E' pertanto necessario, in tali casi, ricordarsi anche di aggiungere direttamente al file BIOUPDATE.TXT corrente un record relativo alla modifica effettuata, cioè un record del tipo B "CCCCCCCCCCCCCCCC_AAAAMMGG_99_0000_ N_0_A<CR><LF>" con i valori aggiornati degli attributi 'N' e/o 'A',

altrimenti le modifiche effettuate non verrebbero applicate a seguito della successiva sincronizzazione con altri terminali “slave”.

- **CANCELLA UTENTE**

  E’ la voce usata per rimuovere tutte le impronte associate ad un certo codice tessera dall’archivio biometrico. X1/X2 chiede solo di leggere il badge dell’utente che deve essere cancellato (timeout: 10 secondi):

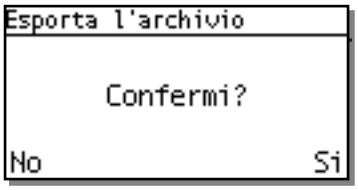


  In alternativa, è anche possibile inserire manualmente il codice utente, in una delle 2 modalità già descritte nel caso della registrazione delle impronte (vedi pag. 124).

  In questo caso non viene chiesta nessuna conferma: X1/X2 controlla la presenza del codice tessera inserito all’interno dei record del file USERCODS.TXT, e se non lo trova mostra il messaggio di errore “**Operazione fallita – Utente non trovato**”. Se lo trova, invece, invalida il corrispondente record sovrascrivendo le prime 10 cifre del campo “codice tessera” con altrettanti caratteri ‘$’=chr(36), controlla nel record lo *shortcode* che era abbinato a quel codice tessera e cancella tutti i *template* identificati da quello *shortcode* che si trovano nella memoria interna del modulo FingerBOX, e infine aggiunge un record al file BIOUPDATE.TXT relativo alla cancellazione di quell’utente, cioè un record del tipo B “CCCCCCCCCCCCCCCC_AAAAMMGG_00_0000_ 0_0_0<CR><LF>” (gli attributi ‘N’ e ‘A’ sono fissi a ‘0’ in quanto irrilevanti per un utente cancellato).

- **ESPORTA L’ARCHIVIO**

  E’ la voce usata per esportare il contenuto corrente dell’intero archivio di impronte (la stessa operazione può anche essere effettuata da remoto mediante il pulsante “**Export archive**” nella pagina “**Biometrics**” del web server HTTP del terminale, come visto al §11 a pag. 120, o tramite caricamento di un file apposito via FTP, vedi §16 a pag. 156).



  Come prima cosa X1/X2 chiede conferma dell’operazione (timeout: 30 secondi):

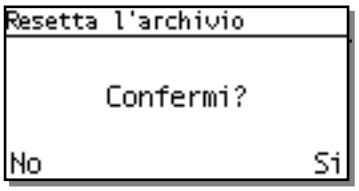
  Premete ↵ (Enter) (Si) per procedere e **[<-]->** (No) per abortire. L’esportazione dell’archivio consiste nella creazione di un file di testo **BIODATA.TXT** all’interno della cartella **\BIOEXP** (normalmente non presente ma creata, assieme alla cartella \BIOIMP, subito dopo avere abilitato la gestione del modulo biometrico FingerBOX).

  Il file **BIODATA.TXT** conterrà dei record aventi lo stesso formato di quelli già visti per il file BIOUPDATE.TXT, ma solo del tipo A: infatti, mentre BIOUPDATE.TXT contiene la “storia” dell’archivio a partire dalla creazione di tale file, BIODATA.TXT è semplicemente una “fotografia” dello stato attuale dell’intero archivio al momento dell’esportazione comandata. Non è rilevante quante e quali operazioni siano state effettuate per arrivare allo stato attuale, pertanto BIODATA.TXT conterrà sempre un solo record per ciascun utente registrato, all’interno del quale si trovano tutti i *template* di quell’utente attualmente contenuti nell’archivio di impronte e gli attuali attributi dell’utente. Se al momento dell’esportazione un precedente file BIODATA.TXT è già presente nella cartella \BIOEXP, esso sarà semplicemente sovrascritto; se l’archivio di impronte è vuoto, il file verrà creato vuoto.

  Lo scopo di BIODATA.TXT è consentire la sincronizzazione (mediante un’unica operazione di copia dell’intero archivio di impronte) dei moduli biometrici FingerBOX di altri terminali X1/X2 “slave” facenti parte dello stesso impianto mediante il semplice trasferimento via FTP (o via chiavetta di memoria USB, vedi §14.5 e §14.6 a pag. 150) di questo file dal terminale “master” agli “slave” (e conseguente importazione automatica dei dati, vedi §11.3 a pag. 134), senza bisogno di ripetere tutte le singole operazioni in locale su ciascuno di essi. A differenza di BIOUPDATE.TXT, dopo la creazione il file BIODATA.TXT non verrà più modificato fino alla successiva esportazione comandata, quindi alla prima variazione dell’archivio non ne rappresenterà più lo stato attuale.

- **RESETTA L’ARCHIVIO**

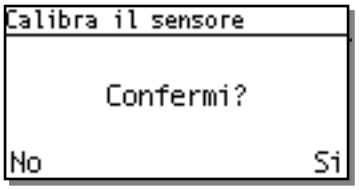
  E’ la voce usata per cancellare l’intero contenuto dell’archivio di impronte (la stessa operazione può anche essere effettuata mediante il pulsante “**Delete all templates**” nella pagina “**Biometrics**” del web server HTTP del terminale, come visto al §11 a pag. 120, o tramite caricamento di un file apposito via FTP, vedi §16 a pag. 156). Come prima cosa X1/X2 chiede conferma dell’operazione (timeout: 30 secondi):



Premete ↵ (Enter) (Si) per procedere e **[<-]->** (No) per abortire. Oltre a tutti i *template* contenuti nella memoria interna del modulo FingerBOX, vengono rimossi il file **USERCODS.TXT** nella *root* del terminale ed il file **BIOUPDATE.TXT** all'interno della cartella **\BIOEXP**, se presente. Un eventuale file BIODATA.TXT già presente nella cartella \BIOEXP non viene invece toccato (come già detto, BIODATA.TXT non rappresenta in generale lo stato attuale dell'archivio, se non nel momento in cui viene creato).

- **CALIBRA IL SENSORE**

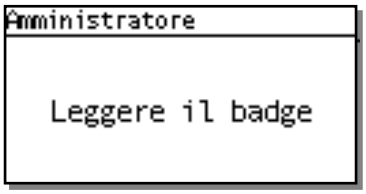
E' la voce usata per calibrare il sensore di impronte digitali (disponibile solo per sensori di tipo capacitivo: per quelli ottici la calibrazione non è necessaria). La stessa operazione può anche essere effettuata mediante il pulsante "**Calibrate Sensor**" nella pagina "**Biometrics**" del web server HTTP del terminale, come visto al §11 a pag. 120). Come prima cosa X1/X2 chiede conferma dell'operazione (timeout: 30 secondi):



Premete ↵ (Enter) (Si) per procedere e **[<-]->** (No) per abortire. Questa operazione non ha alcun effetto sul contenuto dell'archivio di impronte, ma può essere effettuata se si verifica un peggioramento delle prestazioni del sensore oppure, in seguito alla sostituzione del sensore, per ottimizzarne il funzionamento.

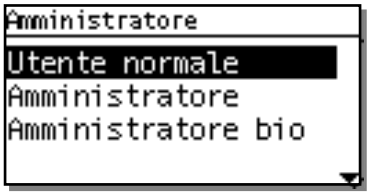
- **AMMINISTRATORE**

E' la voce usata per assegnare ad uno o più utenti gli attributi di "amministratore". Un utente amministratore ha la possibilità di entrare nel menu supervisore (nel caso di amministratore di tipo "generale"), oppure direttamente nella sezione "Biometrics" del menu (nel caso di amministratore di tipo "solo biometrico") semplicemente leggendo la propria tessera ed effettuando la verifica biometrica 1:1, oppure anche con il solo dito (ma solo se la modalità "identificazione 1:N" è abilitata, vedi §11 a pag. 120), senza bisogno di conoscere e digitare la password operatore. Da quanto detto segue che per poter essere definito come amministratore un utente deve avere precedentemente registrato almeno un'impronta. Come prima cosa X1/X2 chiede di leggere il badge dell'utente che si vuole definire come amministratore (timeout: 10 secondi):



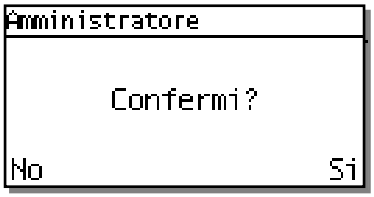
In alternativa, è anche possibile inserire manualmente il codice utente, in una delle 2 modalità già descritte nel caso della registrazione delle impronte (vedi pag. 124).

Qualunque sia il codice tessera inserito, X1/X2 mostra a questo punto la schermata di selezione degli attributi dell'utente (timeout: 30 secondi):



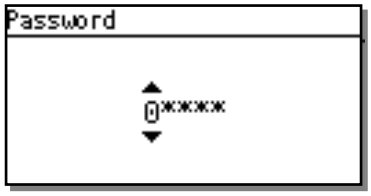
Usate i tasti freccia ▲▼ per selezionare una voce di menu e ↵ (Enter) per confermare. Tutti gli utenti sono inizialmente definiti come "utenti normali". La prima opzione serve quindi esclusivamente per riportare allo stato di utente normale un utente precedentemente definito come amministratore.

Solo una volta effettuata la selezione, X1/X2 controlla la presenza del codice tessera inserito all'interno dei record del file USERCODS.TXT, e se non lo trova mostra il messaggio di errore "**Operazione fallita – Utente non trovato**". In caso contrario, chiede conferma dell'operazione prima di procedere (timeout: 30 secondi):



Premete ↵ (Enter) (Si) per procedere e **[<-]->** (No) per abortire.

Vediamo ora cosa succede dopo avere definito un utente amministratore. Premendo i tasti **.F** + ▲ compare il prompt di richiesta password operatore (timeout: 30 secondi):



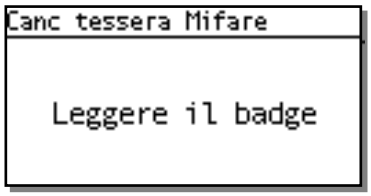
Se non è stato definito nessun amministratore di tipo “generale”, è ancora possibile inserire la password manualmente da tastiera per accedere al menu supervisore. In alternativa, un amministratore del tipo “solo biometrico” può leggere il proprio badge (non è mai possibile in questo caso digitare manualmente il codice tessera), a cui seguirà la richiesta di appoggiare il dito sul sensore per procedere alla verifica biometrica 1:1, a meno che l’utente non sia esentato dalla verifica biometrica su lettura della tessera; solo se la modalità “autoscan” è abilitata (vedi §11 a pag. 120), è anche possibile appoggiare direttamente il dito sul sensore per procedere all’identificazione 1:N. Se l’autenticazione ha esito positivo, è possibile procedere. Nota: un amministratore solo biometrico, non essendo un amministratore generale, può accedere solo alla sezione “Biometrics” del menu supervisore descritta in questo paragrafo (la schermata principale “Configurazione”, vedi §10.5 a pag. 113, non gli viene neppure mostrata), inoltre non ha la facoltà di definire gli attributi di amministratore di un altro utente, pertanto la voce “Amministratore” non gli compare fra le opzioni del menu “Biometrics”.

Tornando al prompt di richiesta password operatore, se è stato definito almeno un amministratore di tipo “generale” non è più possibile inserire la password manualmente da tastiera, anche se corretta (al momento della conferma il prompt di richiesta password viene sistematicamente abortito e il terminale emette 3 beep consecutivi, oltre a registrare un messaggio di errore nel file LOG.TXT). Solo gi amministratori generali possono in questo caso accedere al menu supervisore completo, previa autenticazione biometrica da effettuare con le stesse modalità già descritte (lettura badge più verifica biometrica 1:1, oppure identificazione 1:N con il solo dito). Gli amministratori solo biometrici possono ancora accedere alla sezione “Biometrics” del menu con identiche modalità.

11.2 SALVATAGGIO DELLE IMPRONTE SU CARTE *MIFARE*

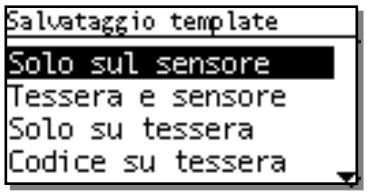
Come visto al §11.1 a pag. 123, nel caso in cui almeno uno dei parametri **CardDecode** all’interno delle sezioni *[Reader1]*, *[Reader2]* e *[ExtReader]* del file PARAMETERS.TXT (vedi §4.11 a pag. 42) sia impostato ai valori ‘30’ / ‘32’ / ‘41’ / ‘42’ / ‘43’ (quelli relativi ad un lettore RFID2/3 seriale TTL 13,56MHz), la sezione “Biometrics” contiene una ulteriore voce “Canc tessera Mifare”, visibile solo spostando la selezione verso il basso fino alla pagina successiva (valgono sempre le note 2 e 3 a pag. 124).

Questa voce può essere utilizzata per cancellare il codice tessera personalizzato e i *template* di un utente precedentemente salvati all’interno di una carta Mifare (vedremo a breve come), rendendo quindi impossibile riutilizzare tale carta ai fini della verifica biometrica su terminali che non contengano già a bordo i *template* di quell’utente. Una volta effettuata la selezione, X1/X2 chiede solo di posizionare sul lettore la carta Mifare che deve essere cancellata (timeout: 10 secondi):



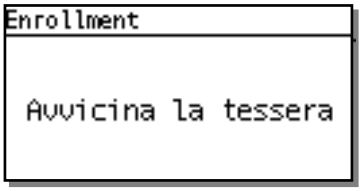
Appena rilevata la carta Mifare, X1/X2 passa alla sovrascrittura di tutti i blocchi dati coinvolti: questo processo richiede alcuni istanti (il terminale mostra il messaggio “**Attendere prego …**” e la percentuale di avanzamento), quindi non si deve allontanare la carta dal lettore fino alla visualizzazione del messaggio di conferma “**Operazione terminata**”. In caso contrario, comparirà il messaggio “**Operazione fallita – Tessera persa**” e la sovrascrittura sarà soltanto parziale.

Come visto al §11.1 a pag. 125, i valori ‘30’ e ‘32’ del parametro **CardDecode** fanno anche sì che prima della schermata per la selezione della modalità di richiesta della verifica biometrica 1:1 ne compaia un’altra (“Salvataggio template”) relativa al salvataggio delle impronte, con la sola eccezione del caso in cui il parametro **TemplateSource** all’interno della sezione *[Biometric]* del file PARAMETERS.TXT (vedi §4.11 a pag. 49) sia impostato ad ‘1’ (in tal caso infatti viene esclusa a priori la ricerca dei *template* su carta: risulta quindi superfluo chiedere all’utente dove debbano essere salvati i *template*, poiché l’unica opzione risulta essere la memoria interna del modulo FingerBOX):



La schermata mostrata a lato (timeout: 30 secondi) si riferisce ai valori '2' (default) e '3' del parametro **TemplateSource**, cioè quelli che prevedono due possibili opzioni per il salvataggio delle impronte (su carta e sul modulo biometrico), seppur con priorità di ricerca differenti. Se invece **TemplateSource** vale '0' (ricerca solo su carta), le opzioni mostrate sono solo 2: "Solo su tessera" e "Codice su tessera".

- Selezionando "**Solo sul sensore**" o "**Tessera e sensore**", il processo di registrazione prosegue come visto al §11.1 a pag. 125, a partire dalla schermata "Verifica biometrica". La differenza, nel caso "**Tessera e sensore**", è che dopo la scansione e prima di passare alla richiesta "Inserire nuovo dito?" viene chiesto di posizionare la carta Mifare sul lettore per procedere al salvataggio dell'impronta sulla tessera (timeout: 10 secondi):



Attenzione: la scrittura dei *template* su carta Mifare richiede alcuni istanti (appena rilevata la carta il terminale mostra il messaggio "**Non rimuovere tessera - Attendere prego …**" e la percentuale di avanzamento), quindi non si deve allontanare la carta dal lettore fino alla visualizzazione della richiesta "Inserire nuovo dito?". In caso contrario, comparirà il messaggio "**Operazione fallita – Tessera persa**" e la scrittura sarà soltanto parziale.

I dati biometrici vengono in tal caso scritti su tutti i blocchi dati adiacenti disponibili a partire da quello specificato dal parametro **MifareFirstBlock** all'interno della sezione *[Biometric]* del file PARAMETERS.TXT (vedi §4.11 a pag. 49): il valore di default è 1, cioè il primo blocco disponibile dopo il blocco 0 a sola lettura che contiene anche il codice fisso UID). Il codice tessera inserito all'inizio della procedura di *enrollment* viene scritto a partire dall'offset 1 (quindi escluso il primo byte) per una lunghezza di 8 byte in formato packed BCD (2 cifre per byte), e ad esso seguono i *template* (si ricordi che per usare il codice tessera personalizzato al posto dello UID in fase di lettura carta è necessario impostare il parametro **CardDecode** nella sezione relativa al lettore utilizzato al valore '32', vedi a pag. 42).

Nota: nonostante nel modulo FingerBOX vengano sempre registrati entrambi i *template* relativi alle due scansioni effettuate per ciascun dito, per ragioni di spazio su una carta Mifare ne viene registrato sempre uno solo dei due (quello relativo alla prima scansione). Inoltre, su una carta Mifare da 1KB è possibile registrare un solo *template*, quindi un solo dito: anche se si registrano 2 dita diverse dello stesso utente sul modulo FingerBOX, sulla carta Mifare sarà comunque presente solo l'ultimo dito registrato. Se si desidera registrare su una carta due diverse dita dello stesso utente è necessario munirsi di carte Mifare da 4KB (nel caso "**Tessera e sensore**" è possibile farlo anche registrando ciascun dito in momenti diversi).

- Selezionando "**Solo su tessera**", invece, il processo di registrazione prosegue subito con la richiesta di appoggiare il dito sul sensore, saltando completamente la schermata "Verifica biometrica": quando si salvano le impronte solo sulla carta, infatti, non è possibile esentare l'utente dalla verifica biometrica 1:1 (non viene neppure creato un record con gli attributi dell'utente, né in USERCODS.TXT né tantomeno in BIOUPDATE.TXT, e non è possibile la digitazione manuale di quel codice tessera). In questo caso, se si desidera registrare su una carta due diverse dita dello stesso utente, oltre a munirsi di carte Mifare da 4KB è necessario effettuare la registrazione di entrambe le dita una dopo l'altra, rispondendo "Si" alla richiesta "Inserire nuovo dito?", senza uscire dalla procedura di *enrollment* dopo la prima registrazione. Se si prova a fare questo dopo avere appena registrato un'impronta su una carta Mifare da 1KB, compare il messaggio di errore "**Operazione Fallita – Limite impronte**".
- Selezionando "**Codice su tessera**", infine, è possibile scrivere su ciascuna carta Mifare solo un codice tessera personalizzato (quello inserito all'inizio della procedura di *enrollment*) diverso dal codice fisso UID, senza dati biometrici. Se si desidera comunque usare la modalità di timbratura "carta + dito", per poter effettuare la verifica biometrica 1:1 occorre salvare i *template* dell'utente sul terminale in un passo successivo, e impostarne la ricerca anche sul terminale. A differenza dei casi precedenti, una volta effettuata questa scelta compare solamente la richiesta di posizionare la tessera sul lettore, ed il processo di scrittura è quasi immediato. Il blocco utilizzato per la scrittura del codice è sempre quello specificato dal parametro **MifareFirstBlock**, e offset, lunghezza e formato del codice sono gli stessi di quando vengono salvati anche i dati biometrici. Anche in questo caso, per usare il codice tessera personalizzato al posto dello UID in fase di lettura carta è necessario impostare il parametro **CardDecode** nella sezione relativa al lettore utilizzato al valore '32', vedi a pag. 42).

11.3 IMPORTAZIONE DI IMPRONTE NEL MODULO BIOMETRICO VIA FTP

Come abbiamo visto al §11.1 a pag. 123, lo scopo del file **BIOUPDATE.TXT** è consentire la sincronizzazione (mediante aggiornamenti in successione) degli archivi di impronte contenuti nei moduli biometrici FingerBOX di altri terminali X1/X2 "slave" facenti parte dello stesso impianto, mediante il semplice trasferimento via FTP (o via chiavetta di memoria USB, vedi §14.5 e §14.6 a pag. 150) di questo file dal terminale "master" agli "slave". Analogamente, lo scopo del file **BIODATA.TXT** è consentire la sincronizzazione (mediante un'unica operazione di copia dell'intero archivio di impronte) dei terminali X1/X2 "slave", sempre mediante il trasferimento via FTP (o via chiavetta di memoria USB) di questo file dal terminale "master" agli "slave". A prescindere da quale tipo di sincronizzazione si scelga, in entrambi i casi il file deve essere copiato via FTP nella cartella **\BIOIMP** di ciascun terminale "slave" (normalmente non presente ma creata, assieme alla cartella **\BIOEXP** , subito dopo avere abilitato la gestione del modulo biometrico FingerBOX). X1/X2 controlla continuamente la cartella **\BIOIMP**, e appena rileva la presenza di uno fra BIOUPDATE.TXT e BIODATA.TXT procede automaticamente all'importazione dei dati biometrici.

Attenzione: nel caso in cui un impianto venga gestito dal programma Xatl@s, la sincronizzazione degli archivi di impronte contenuti nei moduli biometrici FingerBOX di tutti i terminali X1/X2 facenti parte dell'impianto viene completamente gestita dal programma stesso (viene anche a cadere il concetto di "master" e "slave", poiché gli *enrollment* possono essere effettuati su qualunque terminale, vedi §11.5 a pag. 135), pertanto è assolutamente proibito trasferire via FTP (o via chiavetta di memoria USB) i file BIOUPDATE.TXT o BIODATA.TXT dalla cartella \BIOEXP di un qualunque terminale alla cartella \BIOIMP di un altro, e tantomeno copiare il file USERCODS.TXT da un terminale ad un altro (operazione da evitare sempre, a prescindere da come viene gestito l'impianto).

Nel caso di **BIOUPDATE.TXT**, l'importazione consiste nel replicare una alla volta tutte le operazioni già effettuate sul terminale "master" a partire dalla creazione del file (la registrazione di ogni singola impronta, ma anche la cancellazione e il cambiamento degli attributi di un utente), nello stesso ordine con cui sono stati registrati i relativi record. Il file BIOUPDATE.TXT non può essere vuoto (viene creato solo in automatico, e solo in seguito alla generazione di un record), ma se lo fosse semplicemente non succederebbe nulla.

Nel caso di **BIODATA.TXT,** invece, l'importazione consiste nella pura copia di tutti i *template* e degli attributi di ciascun utente presenti nel file (un solo record per ciascun utente, contente tutti i dati di quell'utente). In questo caso l'archivio biometrico del terminale "slave" viene prima cancellato, per essere sicuri che il risultato finale sia l'esatta replica di quanto contenuto nel file. Se il file BIODATA.TXT è vuoto (il che accade quando viene creato mediante l'esportazione comandata di un archivio vuoto), l'archivio del terminale "slave" rimarrà quindi vuoto.

In entrambi i casi, i file BIOUPDATE.TXT e BIODATA.TXT vengono automaticamente rimossi dalla cartella **\BIOMP** subito dopo l'importazione.

Come abbiamo visto, il trasferimento del file USERCODS.TXT dal terminale "master" non è necessario per importare i dati biometrici su uno "slave", anche se è l'unico che contiene l'informazione relativa agli *shortcode* assegnati a ciascun codice tessera. Infatti, al momento di importare un *template* relativo ad un codice tessera non ancora presente nel file USERCODS locale del terminale "slave", a quel codice tessera viene automaticamente riassegnato lo *shortcode* contenuto nel primo record invalidato trovato (se ne esiste uno), oppure un nuovo *shortcode* generato in maniera sequenziale aggiungendo un nuovo record in coda al file. In pratica, gli *shortcode* vengono rigenerati in maniera indipendente su ciascuno "slave" in base al contenuto del file USERCODS.TXT locale, e possono essere diversi fra loro e diversi da quelli del terminale "master" a parità di codice tessera: questo non pregiudica il funzionamento del sistema, infatti gli *shortcode* non compaiono mai nei dati delle transazioni e servono solo localmente per identificare nella memoria interna di ciascun modulo FingerBOX i *template* relativi ad un certo codice tessera.

Vediamo adesso come effettuare l'importazione dei dati biometrici generati da altri tipi di terminali. E' supportata l'importazione da terminali 929 FingerTRAX+G/SU e 962 SuperTRAX con modulo biometrico esterno FingerBOX (lo stesso utilizzato da X1/X2). Su entrambi questi tipi di terminali sono disponibili due modalità di funzionamento: con e senza file **USERCODS**. Nella modalità senza USERCODS i codici tessera utilizzati devono essere uguali agli *shortcode* ad essi associati (max 4 cifre significative): in questo caso l'esportazione dei dati biometrici prevede la generazione ed il trasferimento del solo file **FINGER**. Nella modalità con USERCODS, invece, oltre al file **FINGER** è necessario trasferire anche lo stesso file **USERCODS**. Infatti, a differenza di quanto accade per il file BIODATA.TXT su X1/X2, il file FINGER generato durante l'esportazione dei dati

biometrici non contiene i codici tessera (necessari per la gestione delle transazioni), ma solo gli *shortcode* che identificano i *template* di ciascun utente: è quindi necessario il file USERCODS che permette di risalire ai codici tessera associati agli *shortcode*. Al momento di importare i dati su X1/X2, gli shortcode vengono comunque rigenerati in maniera indipendente in base al contenuto del file USERCODS.TXT locale, come già visto per l'importazione da terminali dello stesso tipo, e possono essere diversi da quelli del terminale di origine a parità di codice tessera.

X1/X2 controlla continuamente la cartella **\BIOIMP**, e appena rileva la presenza di un file chiamato "**FINGER**" (senza estensione) controlla innanzitutto la presenza dell'eventuale file "**USERCODS**" associato (anch'esso senza estensione), e se non lo trova subito controlla altre 3 volte a intervalli regolari; dopo circa 15 secondi, se non ha trovato nessun file USERCODS, procede all'importazione assumendo che il file FINGER sia stato generato nella modalità "senza USERCODS": in questo caso utilizza come codici tessera gli *shortcode* contenuti nel file FINGER, con riempimento di zeri a sinistra. Se invece trova il file USERCODS (caricando entrambi i file insieme il processo parte immediatamente) durante l'importazione vengono usati i codici tessera ivi presenti. In ogni caso, i file FINGER e USERCODS vengono rimossi dalla cartella **\BIOIMP** al termine dell'importazione. Si noti che il file BIOUPDATE.TXT (all'interno della cartella **\BIOEXP**) non viene mai creato né modificato in fase di importazione, anche se in effetti il contenuto dell'archivio è cambiato in seguito a questa operazione.

Nota: vengono correttamente importate anche le impronte registrate nella modalità non-standard "singolo *template* per impronta" (disponibile solo su terminali 929 FingerTRAX+G/SU e 962 SuperTRAX opportunamente configurati), e pure i *template* registrati nel formato non-standard a 256 byte (disponibile solo su terminali 962 SuperTRAX opportunamente configurati, vedi campo "DDDD" nei record dei file BIOUPDATE.TXT e BIODATA.TXT al §11.1 a pag. 128).

11.4 ULTERIORI MODALITA' DI ESENZIONE DALLA VERIFICA BIOMETRICA

Nel caso vi sia la necessità che visitatori temporanei possano effettuare transazioni senza dover procedere alla registrazione delle impronte, è possibile impostare il parametro **FreePass**=1 nella sezione *[Biometric]* del file PARAMETERS.TXT, vedi §4.11 a pag. 49, o analogamente spuntare la *checkbox* "**Visitors free pass**" nella pagina "**Biometrics**" del web server HTTP del terminale, vedi §11 a pag. 120. In tal modo, per ogni lettura (o digitazione, ma solo su X2 e se il parametro **AllowTypeCode**=1 all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT, vedi §4.11 a pag.31) di un codice per il quale non sia già stata effettuata una registrazione di impronte, non verrà richiesta la verifica biometrica: se il codice tessera è valido secondo i criteri di controllo accessi (o in ogni caso se il controllo accessi è disabilitato) la transazione verrà quindi immediatamente accettata.

Esiste anche la possibilità di evitare la richiesta di verifica biometrica non in base al codice tessera letto, ma in base al lettore utilizzato: a tale scopo è disponibile un parametro **SkipBioVerify** nelle sezioni *[Reader1]* e *[ExtReader]* del file PARAMETERS.TXT (vedi §4.11 a pag. 46; la sezione *[Reader2]* non è interessata in quanto viene ignorata in presenza di un modulo biometrico FingerBOX abilitato). Impostando a 1 tale parametro in una o in entrambe queste sezioni, o analogamente spuntando la checkbox "**Skip biometrics verify**" nelle pagine "**Reader 1**" e/o "**External Reader**" del web server HTTP del terminale, non verrà chiesta la verifica biometrica per tutte le letture effettuate sui lettori corrispondenti (***Nota***: le impostazioni "**External Reader**" si applicano anche ad eventuali lettori aggiuntivi collegati su schede di espansione 914 NeoMAX opzionali, vedi §3.6 a pag. 11).

11.5 *ENROLLMENT* DISTRIBUITI SOTTO XATL@S

Nel caso in cui un impianto venga gestito dal programma Xatl@s, viene a cadere il concetto di "master" e "slave", poiché gli *enrollment* possono essere effettuati su qualunque terminale, o anche presso una stazione di *enrollment* su PC. In tale situazione la sincronizzazione degli archivi di impronte contenuti nei moduli biometrici FingerBOX di tutti i terminali X1/X2 facenti parte dell'impianto viene completamente gestita dal programma stesso, sulla base di un database centralizzato che viene popolato progressivamente prelevando i nuovi dati biometrici memorizzati su ciascun terminale. ***Nota***: come regola,

nel caso ad un certo punto risultino essere presenti dati biometrici diversi su terminali diversi relativi allo stesso codice tessera , verranno mantenuti nel database di Xatl@s soltanto i *template* presenti nell'ultimo terminale da cui sono stati acquisiti i dati biometrici di quell'utente*. Pertanto, non ha senso effettuare diversi *enrollment* dello stesso utente su terminali diversi, neanche se relativi a dita diverse dello stesso utente. Ad esempio, se viene effettuato l'*enrollment* di un dito di un utente su un terminale, e successivamente l'*enrollment* di un altro dito dello stesso utente su un altro terminale, nel database di Xatl@s verranno comunque mantenuti solo i *template* relativi all'unico dito presente nell'ultimo terminale da cui sono stati acquisiti i dati biometrici di quell'utente*.

* L'ultimo terminale da cui sono stati acquisiti i dati biometrici di un utente può essere quello su cui è stato effettuato l'ultimo *enrollment* da parte di quell'utente (se tutti i terminali interessati erano già stati "in linea" prima dell'ultimo *enrollment*), ma può anche essere un altro terminale che non era ancora mai stato "in linea" prima dell'ultimo *enrollment*, e che lo è diventato successivamente.

12. TRANSAZIONI ONLINE VIA HTTP

La gestione del protocollo HTTP può essere attivata impostando a 1 il parametro **Protocol** nella sezione *[Ethernet]* del file PARAMETERS.TXT (default 0, vedi §4.11 a pag. 56) oppure, analogamente, selezionando il *radio button* "**HTTP**" fra le opzioni "**Protocol**" nella pagina "**Network**" del web server HTTP.

In questo caso, a seconda del valore del parametro **Offline** nella sezione *[TimeAttendance]* del file PARAMETERS.TXT, §4.11 a pag. 31), X1/X2 invia un messaggio **HTTP GET** al **MasterUrl** (vedi PARAMETERS.TXT) per ogni transazione (in tempo reale se il server è attualmente raggiungibile, o successivamente non appena ritorna in linea), e quindi si mette in attesa di una risposta dal server (**ConnTimeout**, vedi PARAMETERS.TXT): pertanto, ci sarà bisogno di un server web pronto a ricevere i messaggi **HTTP GET** all'indirizzo **MasterUrl**.

Inoltre, se un modulo biometrico FingerBOX esterno è presente e abilitato, per default X1/X2 invia un messaggio **HTTP POST** al **MasterURL** per ciascun nuovo record memorizzato nel file BIOUPDATE.TXT, che può essere relativo a qualunque operazione effettuata all'interno del menu di gestione dell'archivio di impronte: registrazione di ogni nuova impronta (inclusi i dati dei *template*), cancellazione utente, cambio degli attributi di un utente (esenzione dalla verifica biometrica, impostazione dei diritti di amministratore). Anche in questo caso, la **HTTP POST** viene inviata in tempo reale se il server è attualmente raggiungibile, o successivamente non appena ritorna in linea (a meno che il file BIOUPDATE.TXT non sia stato cancellato nel frattempo). Si veda il §12.6 a pag. 145 per ulteriori dettagli. Qualora non fosse necessaria, comunque, questa funzione può essere disabilitata impostando il par. **SendTemplate**=0 nella sezione *[Biometric]* del file PARAMETER.TXT (vedi 4.11 a pag. 50).

12.1 MESSAGGI HTTP PER TRANSAZIONI ONLINE (DA X1/X2 A MASTERURL)

I messaggi di transazione online inviati in tempo reale sono pacchetti HTTP GET costruiti concatenando i seguenti 2 parametri: **MasterUrl** e **httpOnlineMessage**.

MasterUrl contiene l'indirizzo web (URL) del server HTTP sulla macchina host o il suo indirizzo IP e, opzionalmente, la porta usata dall'host (default 80), ad esempio MasterUrl= www.customerserver.com:8181, oppure 192.168.1.200:80

httpOnlineMessage contiene la parte rimanente dell'URL del messaggio HTTP GET che viene inviato al server

Questo parametro può contenere una stringa URL personalizzata, in cui è normalmente presente un identificatore di "tipo messaggio" (ad esempio "*/online*"), oltre ad alcuni o tutti i seguenti "*server tag*", a seconda delle vostre preferenze:

- *$transaction$*
  se presente, viene rimpiazzato col contenuto del record così come verrebbe memorizzato nel file TRANSACTIONS.TXT assumendo che la transazione sia valida (quindi per le transazioni online i campi "CONTROLS" e "RESULT" sono fissi a "00" e "0", rispettivamente); inoltre, solo per le transazioni online, il campo "EDITION" viene sempre aggiunto subito dopo il campo "SORGENTE" (anche nel caso debba rimanere vuoto, il separatore "," dopo il campo SORGENTE è comunque presente).

- *$fullcode$*
  se presente, viene rimpiazzato con l'intero codice restituito dal lettore di badge (limitato a max 80

caratteri), cioè la lettura originaria da cui poi è stato estratto il codice personale in base al valore dei parametri **CardCodeBegin** e **CardCodeLength** della relativa sezione del file PARAMETERS.TXT

- *$termid$*
  se presente, viene rimpiazzato col contenuto del parametro **TermID**
  Nota: gli eventuali caratteri non alfanumerici (come ad esempio gli spazi) contenuti nel valore del par. **TermID** vengono automaticamente convertiti nel relativo codice *percent-encoding*, anche se il par. **EncodeUrl**=0 nella sezione *[Ethernet]* del file PARAMETERS.TXT (vedi §4.11 a pag. 56) .

- *mac*
  se presente, viene rimpiazzato con l'indirizzo MAC del terminale

- *$localip$*
  se presente, viene rimpiazzato con l'indirizzo IP del terminale

- *$dhcp$*
  se presente, viene rimpiazzato con lo stato DHCP (0=DCHP disabilitato, 1=DHCP abilitato)

- *$date$*
  se presente, viene rimpiazzato con la data nel formato AAAAMMGG

- *$time$*
  se presente, viene rimpiazzato con l'ora nel formato HHMMSS

- *$localtransaction$*
  0= il file TRANSACTIONS.TXT non esiste
  1= esiste un file TRANSACTIONS.TXT locale (transazioni registrate in locale a causa di una mancata risposta del server)

- *io*
  se presente, viene rimpiazzato da una stringa che mostra lo stato attuale degli ingressi digitali / relé, inclusi i dispositivi locali e remoti (ovvero quelli su schede di espansione 914 NeoMAX opzionali). Il formato della stringa di stato è il seguente: $\mathbf{I}i_1i_2i_3i_4i_5i_6\mathbf{O}r_1r_2r_3r_4r_5$, dove i_n e r_n sono gli stati degli ingressi *n* e relé *n*, rispettivamente: '0'=non attivo, '1'=attivo, '_'=non disponibile (solo per i dispositivi remoti). Ad esempio, se sono disponibili solo i dispositivi locali ed essi sono tutti non attivi, la stringa di stato è `I00____O0____`

- *$batt$*
  se presente, viene rimpiazzato con l'attuale stato di carica della batteria (vedi §13.1 a pag.147)

- *$battmV$*
  se presente, viene rimpiazzato con l'attuale valore in mV della tensione sulla batteria

Nota: se qualcuno dei server tag sopra menzionati dovesse essere rimpiazzato da una stringa contenente degli spazi, ciascun singolo carattere spazio sarà automaticamente sostituito dalla stringa "**%20**", che ne rappresenta il valore esadecimale in codifica ASCII (gli spazi non sono consentiti all'interno di un URL).

Esempio:

MasterUrl=www.yourserver.com:8181

httpOnlineMessage=/online?badge=$transaction$&TerminalID=$termid$&mac=mac

Quando un utente effettua una transazione il terminale invia il seguente messaggio HTTP GET:

http://www.yourserver.com:8181/online?badge=20101201,152110,0,0,123456789,1&TerminalID=x1maindoor&mac=00:04:24:00:00:00:11:22

Se il terminale non riceve una risposta dal server entro il tempo definito dal parametro **ConnTimeout**, la transazione viene registrata localmente nel file TRANSACTIONS.TXT.

Le successive transazioni verranno immediatamente registrate in locale, finché il server non sarà nuovamente in linea.

Piuttosto che nei messaggi delle transazioni online, il server tag *$localtransaction$* viene tipicamente usato nei pacchetti "*Keep Alive*" (vedi §12.3 a pag. 140), che vengono inviati al server periodicamente per segnalare la presenza del file TRANSACTIONS.TXT.

12.2 FORMATO RISPOSTA DEL SERVER (DA MASTERURL A X1/X2)

Il server risponde ai terminali con una HTTP RESPONSE contenente semplice testo. Questo può essere implementato molto facilmente con i moderni strumenti di programmazione come .NET.

Il messaggio contenuto nella HTTP RESPONSE non è HTML, ma una serie di campi di testo (uno per ciascuna linea, l'ordine non è importante):

screen= <messaggio che sarà visualizzato sullo schermo del terminale>

show= <tempo di visualizzazione del messaggio "*screen*" in secondi> (se non specificato, viene usato il valore corrente del parametro **ShowCode** in PARAMETERS.TXT, vedi §4.11 a pag. 30)

save= 0 → non registra la transazione nella memoria del terminale (la revisione dati locale non sarà possibile)
1 → la transazione viene registrata anche localmente
(questo campo viene ignorato se presente in una risposta ad un messaggio "*Keep Alive*")

beep= <numero di segnalazioni acustiche emesse>

relay= <indice del relé>,<tempo di attivazione del relé in decimi di secondo>

Il campo <indice del relé> può assumere i valori 1 (relé interno), 2 e 3 (relé remoti, solo se è stata collegata una scheda di espansione 914 NeoMAX opzionale con indirizzo 1), 4 e 5 (relé remoti, solo se è stata collegata una scheda di espansione 914 NeoMAX opzionale con indirizzo 2), vedi §3.3 a pag. 9.

relay1= <tempo di attivazione del relé 1 in decimi di secondo>

(formato alternativo valido solo per il relé 1 interno, mantenuto per retrocompatibilità)

time= HHMMSS

date= AAAAMMGG

keepaliveperiod= <intervallo fra due successivi pacchetti *Keep Alive*>

pin=XXXX[,YYYY]

Consente di effettuare una "gestione online" del PIN, svincolata dalla configurazione del controllo accessi in modalità offline (vedi §5 a pag. 68). In seguito ad una transazione online, questo campo attiva una richiesta di introduzione del PIN sul display, oltre a specificare il PIN atteso dal server per quel codice utente (XXXX) e, opzionalmente, il PIN di "accesso sotto minaccia" (YYYY). Se il PIN introdotto dall'utente coincide con XXXX (o, nel caso in cui venga specificato, con YYYY), il terminale genera un nuovo messaggio HTTP online, uguale a quello immediatamente precedente ma accodando il campo "&pin=XXXX" (oppure "&pin=YYYY", a seconda di quale sia stato il PIN inserito) al campo *$transaction$*. In caso contrario il terminale mostra il messaggio "Pincode errato", non genera nessun nuovo messaggio online e non registra comunque alcuna transazione in locale.

Nota: ovviamente non ha senso inserire questo campo nella risposta ad un messaggio "*Batch*", cioè una transazione non ricevuta in tempo reale (vedi §12.5 a pag. 144), o ad un messaggio "*Keep Alive*" (vedi §12.3 qui sotto).

print=<testo da inviare alla stampante><CR><LF>

Se la gestione della stampante è abilitata, come descritto al §3.8 a pag. 13, il testo specificato viene inviato alla stampante. Potete anche includere caratteri speciali o non stampabili, usando la notazione **{***DD***}** (che viene rimpiazzata dal singolo carattere il cui codice ASCII decimale è *DD*): ad esempio, nel caso dobbiate stampare diverse linee potete usare la sequenza di caratteri speciali {13}{10} per andare alla linea successiva.

Nota: non è consentito inserire questo campo nella risposta ad un messaggio "*Batch*" (cioè una transazione che non è stata ricevuta in tempo reale, vedi §12.5 a pag. 144), o ad un messaggio "*Keep Alive*" (vedi §12.3 qui sotto).

Esempio:

screen=\fBenvenuto|Sig. Rossi
beep=1
relay=1,10

Pulisce lo schermo ("\f") e mostra il messaggio "Benvenuto Sig. Rossi" su 2 linee ("|"). Il segnalatore acustico emetterà un suono singolo e il relé interno verrà attivato per 1 secondo.

La stessa HTTP RESPONSE viene anche usata per la risposta ai pacchetti "*Keep Alive*" inviati dall'X1 al MasterURL (vedi prossimo paragrafo). In tale caso, inoltre, è possibile usare due ulteriori campi che non vengono gestiti nella risposta ad una transazione online: "*cmd=*" e "*file=*" (vedi §12.4 qui sotto).

12.3 MESSAGGIO "KEEP ALIVE" (DA X1/X2 A MASTERURL)

X1 e X2 inviano in continuazione i pacchetti HTTP GET "*Keep Alive*" al MasterURL configurato nel file PARAMETERS.TXT

Il formato di questo pacchetto "*Keep Alive*" è ottenuto concatenando i seguenti 2 parametri: **MasterUrl** e **httpKeepAliveMessage**

Il formato del parametro **httpKeepAliveMessage** è lo stesso di **httpOnlineMessage**, con gli stessi *server tag* (vedi §12.1 a pag. 137, l'unico *server tag* che non viene considerato è *$transaction$*) ma con un diverso identificatore di "tipo messaggio" (ad esempio "*/keepalive*").

Ogni 15 secondi (per default, vedi **KeepAliveInterval** nel file PARAMETERS.TXT), X1/X2 effettua una HTTP GET al server.

Lo scopo di questo messaggio è molteplice:

- In uno scenario online, informa il server che ci sono delle transazioni registrate in locale nella memoria del terminale (il server tag *$localtransaction$* deve essere incluso in **httpKeepAliveMessage**)
- In uno scenario con reti complesse (firewall, NAT e / o GPRS) il server deve mettersi in attesa del pacchetto "*Keep Alive*" per scoprire l'indirizzo IP dinamico e la porta sorgente del terminale, che deve essere noto per poter stabilire una connessione col terminale (in questo caso il server tag *$termid$* in **httpKeepAliveMessage** è molto importante per identificare il terminale)
- In generale, il server può inviare a X1/X2 alcuni comandi di tipo "*shell*" solo in risposta alla ricezione del pacchetto "*Keep Alive*", come si vedrà al §12.4. Il server può anche "forzare" in qualunque momento l'invio immediato di un pacchetto "*Keep Alive*", proprio allo scopo di eseguire un comando in tempo reale: è sufficiente effettuare una HTTP GET all'URL **http://*<Indirizzo_IP_Terminale>*/keepalive_req.cgi**

Nota: la funzionalità "Keep Alive" via HTTP è indipendente e si aggiunge senza sostituirla alla funzionalità "Keep Alive" via UDP descritta al §3.5 a pag. 10: seppur apparentemente simili, quest'ultima ha uno scopo più limitato in quanto il terminale la usa solo segnalare la sua esistenza e farsi identificare, ma in ogni caso non si aspetta una risposta.

12.4 FORMATO RISPOSTA DEL SERVER AL "KEEP ALIVE" (DA MASTERURL A X1/X2)

Il server risponde ai pacchetti "*Keep Alive*" ricevuti dal terminale con pacchetti dello stesso formato di quelli usati per rispondere alle transazioni HTTP online.

Il pacchetto è ancora una HTTP RESPONSE contenente semplice testo (no HTML).

Le differenze sono le seguenti: 1) in questo caso i campi *"save"* e "*print*" non hanno senso e quindi vengono ignorati; 2) il campo "*pin=*" non ha senso e quindi non deve essere usato; 3) in questo caso è possibile usare alcuni ulteriori campi (che verrebbero gestiti allo stesso modo anche nella risposta ad una transazione online, ma che in tal caso non avrebbero molto senso, ad eccezione del campo "cmd=RDR…" descritto qui sotto):

gprs=off

Questo campo consente di forzare la chiusura dell'attuale connessione GPRS (se ne è attualmente in corso una). Utile per le connessioni GPRS schedulate (vedi §15 a pag.152).

cmd=<CMD> [<PARAM1> … <PARAMn>]

dove <CMD> è un comando da eseguire e i campi fra parentesi sono parametri opzionali il cui numero e significato dipendono dal tipo di comando.

Per ciascun comando inviato in risposta ad un "*Keep Alive*", X1/X2 risponde immediatamente al server con un nuovo pacchetto HTTP GET "*Keep Alive*" a cui aggiunge, in coda, la stringa "*&cmd=ok*" oppure "*&cmd=error*" a seconda dell'esito.

La lista dei comandi disponibili e relativi parametri è la seguente (note: se sono presenti degli spazi all'interno di un parametro è sempre necessario che tale parametro sia delimitato da virgolette; il campo <CMD> deve contenere solo lettere maiuscole):

RA <record> <nome file>

Aggiunge un record al file specificato. Nota 1: <record> deve avere lo stesso numero di caratteri di tutti gli altri record già presenti (file con record a lunghezza fissa). Nota 2: se il file non termina con la sequenza di caratteri <CR><LF>,la stringa <record> sarà semplicemente accodata all'ultimo record già esistente senza aggiungere alcuna linea al file. Nota 3: se il file non esiste viene automaticamente creato con <record> come prima linea. Nota 4: non viene controllato se il file contiene già uno o più <record> identici, quindi nel caso ne viene comunque aggiunto uno ulteriore.

RD <chiave> <nome file>

Cancella (invalida) tutti i record trovati nel file specificato la cui parte iniziale coincide con la stringa fornita come chiave, sostituendoli con la stringa fissa "$$$$$$$$$$$$$$$$$$$$"(20 caratteri '$'=chr(36)). Nota: affinché non venga alterata la struttura del file, questo comando può essere utilizzato soltanto per i file con record a lunghezza fissa >=20.

RM <chiave> <nome file> <nuovo record>

Cancella (invalida) tutti i record trovati nel file specificato la cui parte iniziale coincide con la stringa fornita come chiave, e sostituisce il primo record invalidato con un unico <nuovo record> (in pratica è equivalente ad un comando **RD** seguito da un comando **RA**). Nota 1: il <nuovo record> deve avere lo stesso numero di caratteri di quello da modificare (file con record a lunghezza fissa). Nota 2: Se non trova nessuna corrispondenza aggiunge comunque una linea contenente <nuovo record> alla fine del file.

DEL <nome file>

Cancella il file specificato. Si può fornire un nome completo oppure un nome senza estensione: solo in quest'ultimo caso viene automaticamente aggiunta l'estensione ".TXT" prima di procedere alla cancellazione.

CONSIDLE [<messaggio> [<tempo in secondi>]]

Sospende l'attività di console del terminale mostrando il messaggio "Non disponibile" (se non viene specificato nessun parametro opzionale), oppure l'eventuale messaggio personalizzato fornito come primo parametro. L'eventuale secondo parametro consente di impostare il tempo di permanenza del messaggio (espresso in secondi): se è assente il messaggio rimane a video per un tempo indefinito, fino all'invio di un successivo comando OFFLINE (vedi sotto).

OFFLINE

Fa uscire il terminale dallo stato "CONSIDLE", tornando alla normale operatività. Nota: NON cambia la modalità online/semi-online/online impostata dal parametro Offline all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.11 a pag. 31).

RESTART

Effettua un riavvio del sistema.

RECOVER gg mm aaaa

Forza la ritrasmissione immediata, in modalità "*batch*" (vedi §12.5 a pag. 144), di tutte le transazioni precedentemente effettuate e registrate in locale a partire dalla data indicata (inclusa). Nota 1: come nel caso della funzione di revisione dati di presenza (vedi §10.7 a pag. 118) è irrilevante se il file TRANSACTIONS.TXT sia stato nel frattempo cancellato oppure no , perché il terminale conserva sempre una copia delle transazioni effettuate nel file riservato **btransactions.loc** (vedi §7 a pag. 96).

Quando si invia questo comando in risposta ad un "*Keep Alive*", X1/X2 risponde subito al server con un nuovo pacchetto HTTP GET "*Keep Alive*" a cui aggiunge in coda, solo in caso di successo e subito dopo la consueta stringa di conferma "*&cmd=ok*", anche la stringa "*&recover=<n>*", dove *<n>* è il numero di transazioni che verranno ritrasmesse, una dopo l'altra, a partire dal pacchetto HTTP GET di tipo "*batch*" immediatamente successivo.

GETPAR <sezione> <parametro>

Richiede il valore di un parametro di configurazione contenuto nel file PARAMETERS.TXT (vedi §4.11 a pag. 30). E' necessario specificare anche il nome della sezione (tra parentesi quadre oppure senza) oltre a quello del parametro: in entrambi i casi potete usare caratteri maiuscoli o minuscoli (è irrilevante, purché il nome sia corretto e non contenga spazi).

Nota: esistono 2 varianti che in realtà non fanno riferimento ad una vera e propria sezione del file PARAMETERS.TXT, ma che possono essere utilizzate per richiedere la versione del firmware dell'eventuale modulo biometrico esterno FingerBOX ed il numero dei *template* attualmente registrati nell'archivio biometrico all'interno del modulo (si tratta di informazioni disponibili anche nella pagina "**Biometrics**" del web server HTTP del terminale, vedi §11 a pag. 120). La sintassi esatta di tali varianti è, rispettivamente, "**GETPAR [Info] BioFW**" e "**GETPAR [Info] BioTemplate**".

Quando si invia questo comando in risposta ad un "*Keep Alive*", X1/X2 risponde subito al server con un nuovo pacchetto HTTP GET "*Keep Alive*" a cui aggiunge in coda, subito prima della consueta stringa di conferma "*&cmd=ok*", anche la stringa "*&parVal=<sezione>,<parametro>,<valore>*", dove *<sezione>* e *<parametro>* coincidono con le stringhe specificate come argomenti del comando (senza le eventuali parentesi quadre per la sezione), mentre *<valore>* è il corrispondente valore attuale. Nota: se a causa di un errore di sintassi il parametro non è stato trovato, quest'ultimo campo rimarrà vuoto.

Nota: gli eventuali caratteri non alfanumerici (come ad esempio gli spazi) contenuti nei valori dei parametri restituiti vengono automaticamente convertiti nel relativo codice *percent-encoding*, anche se il par. **EncodeUrl**=0 nella sezione *[Ethernet]* del file PARAMETERS.TXT (vedi §4.11 a pag. 56) .

SETPAR <sez1><TAB><TAB><par1>=<val1>[<TAB><TAB><par2>=<val2> …]
[<TAB><TAB><sez2><TAB><TAB><par1>=<val1>[<TAB><TAB><par2>=<val2> …] …]

Imposta contemporaneamente uno o più parametri, all'interno di una o più sezioni del file PARAMETERS.TXT (vedi §4.11 a pag. 30), ai valori specificati. E' necessario specificare anche i nomi delle

sezioni (sempre tra parentesi quadre) oltre a quelli dei parametri: in entrambi i casi potete usare caratteri maiuscoli o minuscoli (è irrilevante, purché i nomi siano corretti e non contengano spazi).

Quando si invia questo comando in risposta ad un "*Keep Alive*", X1/X2 risponde subito al server con un nuovo pacchetto HTTP GET "*Keep Alive*" a cui aggiunge in coda solo la consueta stringa di conferma "*&cmd=ok*".

BIOADD R_CCCCCCCCCCCCCCCC_AAAAMMGG_nn_DDDD_ttt...ttt_N_T_A_M_B_ff_E<CR><LF>

oppure

BIOADD R_CCCCCCCCCCCCCCCC_AAAAMMGG_nn_DDDD_ttt...ttt_N_T_A_M_B<CR><LF>

Causa l'importazione immediata dei dati biometrici specificati. Rispetto al formato dei record del file BIOUPDATE.TXT (tipo A o tipo B, vedi §11.1 a pag. 128, a seconda che il record sia relativo ad una registrazione di impronta o ad un cambio degli attributi di un utente / registrazione di un utente senza *template* (in quanto esentato dalla verifica biometrica) – per la cancellazione di un utente si deve usare il successivo comando BIODEL), le differenze sono le seguenti: 1) il campo R ha lo stesso significato ma è spostato all'inizio; 2) nel caso di cambio degli attributi di un utente, DDDD è fisso a "0000", nn è fisso a "99" e il campo ttt…ttt è assente, ma entrambi i delimitatori "_" prima e dopo il campo mancante devono essere comunque presenti; per retrocompatibilità, i campi ff e E possono anche essere omessi (in questo caso, se il campo ttt…ttt è presente, esso viene considerato come espresso in formato standard oppure crittografato solo in base al valore corrente del parametro **CryptoEnabled** nella sezione *[System]* del file PARAMETERS.TXT, vedi §4.11 a pag. 53). Il record deve sempre essere terminato con <CR><LF>.

Note: questo comando ha effetto solo se è presente un modulo biometrico FingerBOX esterno, ma non ritorna alcun errore anche nel caso non lo sia. Nel file **BIOUPDATE.TXT**, che tiene traccia solo delle operazioni effettuate all'interno del menu di gestione dell'archivio biometrico, non vi sarà traccia dell'operazione di importazione effettuata tramite questo comando.

BIODEL R_CCCCCCCCCCCCCCCC

Cancella l'utente con codice tessera CCCCCCCCCCCCCCCC (sempre 16 cifre, con eventuale riempimento di zeri a sinistra) e che ha effettuato la registrazione di impronte inserendo il codice tessera mediante il lettore R, che può assumere solo i valori '1' e '3'. Tale valore viene confrontato col corrispondente campo R all'interno dei record del file USERCODS.TXT (vedi §11.1 a pag. 127): se quest'ultimo dovesse valere '0', l'utente verrà cancellato comunque. ***Note***: questo comando ha effetto solo se è presente un modulo biometrico esterno FingerBOX, ma comunque non dà errore in caso contrario. Nel file **BIOUPDATE.TXT**, che tiene traccia delle sole operazioni effettuate all'interno del menu di gestione dell'archivio di impronte, non rimane traccia delle operazioni di cancellazione utente effettuate tramite questo comando.

BIORESET

Cancella l'intero contenuto dell'archivio di impronte (la stessa operazione può anche essere effettuata mediante il pulsante "**Delete all templates**" nella pagina "**Biometrics**" del web server HTTP del terminale, vedi §11 a pag. 120). ***Note***: questo comando ha effetto solo se è presente un modulo biometrico esterno FingerBOX, ma comunque non dà errore in caso contrario. Vengono rimossi sia il file **USERCODS.TXT** che **BIOUPDATE.TXT**.

RDR <numero del lettore> <comando per il lettore>

Invia il <comando per il lettore> specificato (che non deve contenere spazi), destinato esclusivamente ad un modulo 13,56MHz RFID2/3 Mifare R&W configurato con il <numero del lettore> specificato: 1, 2 o 3 (per il "EXTERNAL READER").

Quando inviate questo comando in risposta ad un "*Keep Alive*" (o ad una transazione online, il che ha senso per applicazioni di gestione di un credito a scalare), X1/X2 risponde immediatamente al server con una nuovo pacchetto HTTP GET "*Keep Alive*" a cui aggiunge in coda, subito prima della consueta stringa di conferma "*&cmd=ok*", anche la stringa "*&rdrReply=<reader reply>*", dove *<reader reply>* è la stringa restituita dal lettore (lasciata vuota in caso di errore nell'invio del comando).

Nota: gli eventuali caratteri non alfanumerici (come ad esempio gli spazi) contenuti nella stringa restituita dal lettore vengono automaticamente convertiti nel relativo codice *percent-encoding*, anche se il par. **EncodeUrl**=0 nella sezione *[Ethernet]* del file PARAMETERS.TXT (vedi §4.11 a pag. 56) .

file=<nome file>,<dimensione file in bytes><CR><LF><contenuto del file, una nuova linea per ogni record>

Questo campo consente di caricare un file sul terminale, e viene seguito dal contenuto del file a partire dalla linea seguente, una linea per ogni record. Nota: la dimensione in bytes deve includere TUTTI i caratteri, inclusi i <CR> e <LF> alla fine di ciascuna linea. Specificando sempre la dimensione corretta è anche possibile inserire più di un campo *file* (ed in qualunque posizione) all'interno della stessa risposta ad un "*Keep Alive*".

Per ciascun file caricato in risposta ad un "*Keep Alive*" X1/X2 risponde immediatamente al server con un nuovo pacchetto HTTP GET "*Keep Alive*" a cui aggiunge, in coda, la stringa "*&file=ok*" (se tutti i file specificati sono stati caricati correttamente) oppure "*&file=error*" (se uno o più file non sono stati caricati correttamente): ovviamente questo non vale se viene caricato un file di aggiornamento firmware valido per il terminale in oggetto (con estensione ".bin", vedi §9 a pag. 107), poiché in tal caso il terminale si riavvia immediatamente per effettuare l'aggiornamento, il che causa un reset della macchina a stati per la gestione delle comunicazioni con il server.

Nota: se qualcuna delle risposte ai comandi sopra menzionati (quelle che cominciano con il carattere '&') dovesse contenere degli spazi, ciascun singolo carattere spazio sarà automaticamente sostituito dalla stringa "**%20**", che ne rappresenta il valore esadecimale in codifica ASCII (gli spazi non sono consentiti all'interno di un URL).

Esempi:

cmd=RA 12345678901234567890 PROVA

cmd=RM 1234 PROVA 11223344556677889900

cmd=RD 1122 PROVA

cmd=DEL PROVA

cmd=CONSIDLE

cmd=CONSIDLE "Attesa configurazione…"

cmd=OFFLINE

cmd=RESTART

cmd=CONSIDLE "Attendere 10 secondi" 10

cmd=RECOVER 1 12 2011

cmd=GETPAR [TimeAttendance] DirMode

*cmd=SETPAR [TimeAttendance]*TAB TAB*DirMode=3*TAB TAB*[Reader1]*TAB TAB*CardDecode=30*TAB TAB*CardCodeLength=10*

cmd=BIOADD 1_0000000000123456_20130918_99_0000__1_0_0_0_0

cmd=BIODEL 1_0000000000123456

cmd=BIORESET

cmd=RDR 1 v

file=PROVA,14
12345
67890

12.5 MODALITA' ONLINE: SERVER NON IN LINEA

In modalità online X1/X2 invia un messaggio HTTP GET al MasterURL ad ogni transazione, come spiegato nel §12.1 a pag. 137.

Se il server non risponde inviando un pacchetto HTTP RESPONSE entro il **ConnTimeout**, il terminale registra le transazioni localmente, ed inizia a lavorare in modalità offline.

A partire da questo momento:

1) I pacchetti "*Keep Alive*" vengono comunque inviati periodicamente al MasterURL. In questa fase tali pacchetti possono anche segnalare la presenza di transazioni nella memoria locale del terminale (file TRANSACTIONS.TXT): a questo scopo il server tag *$localtransaction$* deve essere incluso nel parametro **httpKeepAliveMessage**.

   Quando il server torna in linea dovrebbe dapprima rispondere al "*Keep Alive*", quindi può scaricare le transazioni via FTP e cancellare il file, oppure riceverle via HTTP come spiegato al seguente punto 2)

2) Appena ricevuto il pacchetto HTTP RESPONSE di risposta al "*Keep Alive*" (il fatto che sia stato usato il server tag *$localtransaction$* è irrilevante per quanto segue), il terminale si rende conto che il server è di nuovo in linea, e quindi inizia subito ad inviare una HTTP GET con la più vecchia transazione registrata in locale e ancora *pendente* (cioè non ancora marcata come "*già ricevuta dal server*" al MasterURL, ma questa volta concatenandogli il parametro **httpBatchMessage**.
   Questo parametro ha lo stesso formato di **httpOnlineMessage** ma un diverso identificatore di "tipo messaggio" (ad esempio "*/batch*"), e viene usato solo per trasmettere le transazioni precedentemente registrate in locale, a partire dal momento in cui il server HTTP non ha risposto in modalità online.

   Esempio:

   httpBatchMessage=/batch?trsn=$transaction$&id=$termid$

   MasterUrl=www.yourserver.com:8181

   Appena ricevuta la risposta al "*Keep Alive*", X1/X2 invia un pacchetto HTTP GET al MasterURL, con la prima transazione precedentemente registrata in locale in modalità offline:

   http://www.yourserver.com:8181/batch?trsn=20101201,152110,0,0,123456789,1&id=x1maindoor

   Il server dovrebbe replicare a questo pacchetto HTTP GET con un pacchetto HTTP RESPONSE di conferma di avvenuta ricezione, che deve necessariamente contenere la linea seguente:

   ack=1

   A questo punto X1/X2 marca la transazione appena inviata in modalità "*batch*" come "*già ricevuta dal server*", e passa alla trasmissione delle successive transazioni registrate in locale in modalità offline, una ad una, con lo stesso formato (usando **httpBatchMessage**), fino alla conferma di ricezione dell'ultima rimasta da parte del server.

Nota 1: Rispondere ad una transazione offline ricevuta in modalità "batch" con una HTTP RESPONSE priva della conferma di avvenuta ricezione (*ack=1*) non ha molto senso: ciò causerebbe semplicemente la continua ritrasmissione della stessa transazione da parte del terminale.

Nota 2: Rispondere ad una transazione offline ricevuta in modalità "batch" con una HTTP RESPONSE contenente la richiesta di introduzione PIN (*pin=*) o un testo da stampare (*print=*) non ha ovviamente senso.

Nota 3: Se in qualunque momento il server dovesse smettere di mandare i pacchetti HTTP RESPONSE di conferma di ricezione dell'ultima transazione offline inviata in modalità "*batch*", o quelli di risposta ai pacchetti "*Keep Alive*" (in entrambi i casi entro il **ConnTimeout** dall'ultimo pacchetto inviato dal terminale), il terminale tornerebbe automaticamente in modalità offline, registrando eventuali nuove transazioni in locale, e inviando periodicamente i soli pacchetti "*Keep Alive*". Alla prima nuova risposta al "*Keep Alive*", il processo riparte come dal punto 2).

12.6 MESSAGGI HTTP ONLINE SU MODIFICA DELL'ARCHIVIO BIOMETRICO (DA X1/X2 A MASTERURL)

I messaggi relativi all'aggiornamento dell'archivio biometrico sono sempre pacchetti HTTP POST costruiti concatenando i parametri **MasterUrl** e **httpBatchMessage** (sia che vengano inviati in tempo reale oppure successivamente).

Come abbiamo visto nel paragrafo precedente, **httpBatchMessage** ha lo stesso formato di **httpOnlineMessage**, ma un diverso identificatore di "tipo messaggio" (ad es. "*/batch*"). Il server tag *$transaction$*, che viene normalmente incluso in **httpBatchMessage** per consentire l'invio delle transazioni offline in modalità "batch", in questo caso viene lasciato vuoto. Il corpo del messaggio HTTP POST contiene del testo semplice nel seguente formato:

*bioupdate=IIIIIIIIII_*<record come memorizzato in BIOUPDATE.TXT>

dove "*IIIIIIIIII*" sono 10 caratteri che rappresentano l'identificatore unico dell'utente associato al codice tessera contenuto nel record di BIOUPDATE.TXT (vedi §11.1 a pag. 128). L'identificatore dell'utente viene preso dal file CARDS.TXT, se presente (vedi §5.4 a pag. 71), altrimenti questo campo viene riempito di zeri.

Il server dovrebbe a questo punto rispondere a questo pacchetto HTTP POST entro **ConnTimeout**, con un pacchetto HTTP RESPONSE di conferma ricezione che deve contenere la seguente linea:

ack=1

Quindi X1/X2 marca il record appena trasmesso (che comunque viene sempre memorizzato in BIOUPDATE.TXT) come "*già ricevuto dal server*". Altrimenti, se il server non è raggiungibile, X1/X2 smette di inviare messaggi online relativi all'archivio biometrico, fino a che il server non sarà di nuovo in linea. Appena questo accade, X1/X2 invia subito una HTTP POST con il più vecchio record di BIOUPDATE.TXT ancora *pendente* (cioè non ancora marcato come "*già ricevuto dal server*") al MasterURL, e quindi trasmette tutti i record successivi, uno alla volta, fino a che non riceve dal server la conferma di ricezione dell'ultimo record rimasto.

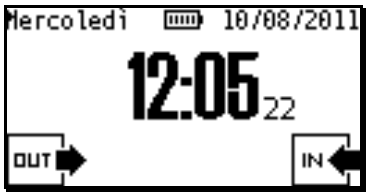
Nota 1: rispondere ad un messaggio online relativo all'archivio biometrico con una HTTP RESPONSE priva della conferma di ricezione (*ack=1*) non ha molto senso: ciò causerebbe semplicemente la continua ritrasmissione dello stesso messaggio da parte del terminale.

Nota 2: se il file BIOUPDATE.TXT viene cancellato mentre il server è fuori linea, allora tutti i record precedentemente memorizzati e ancora *pendenti* non saranno mai ricevuti dal server.

Nota 3: qualora non fossero necessari, i messaggio online relativi all'archivio biometrico possono essere disabilitati impostando il par. **SendTemplate**=0 nella sezione *[Biometric]* del file PARAMETER.TXT (vedi 4.11 a pag. 50).

13. FUNZIONAMENTO A BATTERIA

X1 e X2 dispongono di una batteria 4,8V 600mAh NiMh che ne consente il funzionamento per un tempo limitato anche in assenza di alimentazione. Durante il funzionamento a batteria, sul bordo superiore del display compare un indicatore di carica (simile a quelli tipici dei telefoni cellulari) che normalmente è assente a terminale alimentato(*):



In queste condizioni, normalmente, la retroilluminazione del display viene spenta per consentire un minore consumo e quindi una maggiore autonomia: fino a 2 ore di funzionamento continuo in stand-by con un singolo lettore RFID 125KHz collegato, anche frazionabile mediante spegnimento manuale o automatico (in caso di inattività, vedi parametro **TurnOffTimeout** all'interno della sezione *[System]* del file PARAMETERS.TXT, §4.11 a pag.50, default 10 minuti) e successiva riaccensione del terminale.

E' comunque possibile fare in modo che lo schermo rimanga retroilluminato anche durante il funzionamento a batteria, limitandone ovviamente l'autonomia che viene così ridotta a soli 90 minuti in stand-by con un singolo lettore RFID 125KHz collegato: è sufficiente impostare il parametro **TurnoffBackLight** all'interno della sezione *[System]* del file PARAMETERS.TXT (vedi §4.11 a pag.50) al valore 0 (il default è 1), o deselezionando la checkbox "**Turn Off Backlight on Battery**" nella pagina "**System**" del web server http.

(*) Nota: solo in presenza di un modem GPRS attivato (§15.1 a pag. 155), l'indicatore di carica non viene mostrato per consentire di visualizzare, nella stessa posizione, l'icona relativa allo stato del modem GPRS.

13.1 RICARICA RAPIDA DELLA BATTERIA

A partire dalla versione di hardware **006** (vedi §3.9 a pag.13) X1 e X2 dispongono della funzione di ricarica rapida, che si attiva automaticamente dopo pochi minuti da quando si torna a fornire alimentazione ad un terminale che in precedenza ha funzionato a batteria, oppure quando il terminale al riavvio non trova il file BATTERY.TXT (che contiene sempre l'ultimo stato della batteria prima dello spegnimento), ad esempio perché è stato cancellato.

La ricarica rapida dura sempre almeno 10 minuti (stato "*Conditioning*"), dopodiché prosegue (stato "*FastCharge*") per un massimo di 18 ore, ma può interrompersi molto prima non appena viene riscontrata una variazione negativa della tensione sulla batteria pari ad almeno 20mV. Alla fine della ricarica rapida, il terminale si riporta nel normale stato di mantenimento della carica della batteria ("*Trickle Charge*").

Tutti i cambiamenti di stato relativi alla carica della batteria vengono registrati nel file BATTLOG.TXT assieme ai valori della tensione di batteria e di alimentazione espressi in mV, che vengono comunque registrati ogni 10 minuti anche in assenza di cambiamenti di stato. Inoltre, il valore attuale della tensione di batteria e lo stato della batteria sono mostrati nella pagina "**System**" del web server http.

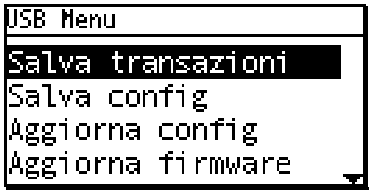
14. UTILIZZO DELLA CHIAVETTA USB

A partire dalla versione di hardware **006** (vedi §3.9 a pag.13) X1/X2 dispongono della funzione di trasferimento automatico di file da/verso chiavetta di memoria USB. Per attivarla, è necessario impostare il parametro **Enable** all'interno della sezione *[USB]* del file PARAMETERS.TXT al valore 1, vedi §4.11 a pag. 59, oppure selezionare la checkbox "**Enable Mass Storage Host**" nella pagina "**USB**" del web server http. Nel caso in cui non sia possibile accedere al terminale via Ethernet neppure per la prima configurazione (e proprio per questo motivo si voglia appunto usare la funzionalità di scarico delle transazioni su chiavetta USB) potete anche usare la voce "USB" all'interno del menu supervisore direttamente dalla console del terminale (vedi §10.5 a pag. 113).

Avvertenza: raccomandiamo di non abilitare questa funzionalità su terminali con versione di hardware precedente alla **006**, perché il risultato potrebbe essere il blocco di tutte o alcune funzioni del terminale, con conseguente necessità di rimuovere la scheda SD e accedervi da un PC per reimpostare il valore di default (0) del suddetto parametro **Enable** all'interno della sezione *[USB]* del file PARAMETERS.TXT e rendere di nuovo utilizzabile il terminale.

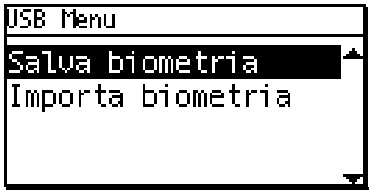
Una volta attivata questa funzionalità, inserendo una chiavetta di memoria USB (che deve essere formattata in FAT32) nell'apposito connettore dopo avere rimosso il cappuccio in gomma (vedi figura 4 al §3.1, pag.7), comparirà una schermata di richiesta password identica a quella per l'accesso al menu supervisore (vedi §10.5 a pag.113), con la differenza che in questo caso la password da inserire deve coincidere con quella specificata dal parametro **PasswordUSB** all'interno della sezione *[USB]* del file PARAMETERS.TXT (default 00000).

Se la password inserita è corretta, apparirà la seguente schermata di selezione "USB Menu":



Questa schermata non ha timeout né possibilità di abortire se non rimuovendo la chiavetta USB. Usate i tasti freccia ▲▼ per selezionare una voce di menu e ↵ (Enter) per confermare.

Solo se X1/X2 è equipaggiato con un modulo biometrico esterno FingerBOX per la scansione di impronte digitali, è possibile che oltre alle voci visualizzate nella schermata riportata sopra ne siano disponibili anche altre due, visibili solo spostando la selezione verso il basso fino alla pagina successiva:



Si veda ai §14.5 e §14.6 a pag. 150 quali sono le condizioni affinché queste voci siano disponibili.

14.1 SCARICO TRANSAZIONI SU CHIAVETTA USB

La prima voce del menu "**Salva transazioni**" serve per trasferire le transazioni precedentemente registrate in offline nel file TRANSACTIONS.TXT sulla chiavetta USB. Le transazioni possono essere "spostate" a tutti gli effetti, il che significa che il file TRANSACTIONS.TXT corrente verrà anche rimosso dal terminale (default), rinominandolo "TRANSACTIONS.0.TXT" e creandone uno nuovo secondo il meccanismo descritto al §4.11 a pag. 32 relativamente al funzionamento del parametro **DeleteOld**=1, oppure semplicemente "copiate" (mantenendole anche sul terminale all'interno del file TRANSACTIONS.TXT): a questo scopo è necessario impostare a 0 il parametro **MoveTrnsToUSB** all'interno della sezione *[USB]* del file

PARAMETERS.TXT (default 1), oppure deselezionare la checkbox "**Move transactions on USB**" nella pagina "**USB**" del web server http. Il nome del file contenente le timbrature che verrà creato sulla chiavetta USB può essere modificato a piacimento mediante il parametro **TrnsFileUSB** all'interno della sezione *[USB]* del file PARAMETERS.TXT (il default è "TRANSACTIONS.TXT" come sul terminale).

Selezionando la voce "**Salva transazioni**" compare una richiesta di conferma:

Pulsanti da utilizzarein questo stato:

Tasti **Clr** oppure **[<-]->** → Abortisce

Tasto ↵ → Conferma

Una volta data conferma, il terminale procede col trasferimento delle transazioni, e dopo breve tempo visualizza il messaggio "Operation Completed", per poi tornare alla schermata "USB Menu".

14.2 SALVATAGGIO CONFIGURAZIONE SU CHIAVETTA USB

La seconda voce del menu "**Salva config**" serve invece per copiare tutti i file presenti nella memoria del terminale (senza distinzioni) in una cartella "\BACKUP" che viene automaticamente creata nella *root* della chiavetta USB. Anche per questa operazione compare la finestra di richiesta conferma già descritta in precedenza, e al termine si torna alla schermata "USB Menu". Logicamente il tempo necessario per la copia risulta maggiore rispetto al caso precedente.

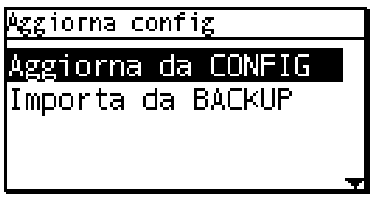
14.3 CARICAMENTO CONFIGURAZIONE DA CHIAVETTA USB

La terza voce del menu "**Aggiorna config**" consente di caricare una determinata configurazione sul terminale (una parte oppure tutti i parametri, e tutti i file necessari) prendendola da una cartella situata nella *root* della chiavetta USB, che può chiamarsi "\CONFIG" oppure "\BACKUP".

Lo scopo di questa distinzione è il seguente: la cartella "\CONFIG" viene utilizzata per effettuare la configurazione di base del terminale (normalmente mediante i file elencati ai §4 e 5, ma anche caricando dei file personalizzati), magari usando la stessa chiavetta USB per impostare in sequenza, uno dopo l'altro, un certo numero di terminali appena installati. I file riconosciuti che non sono finalizzati all'impostazione del dispositivo (ad esempio tutti i file di log e i file contenenti transazioni) non vengono caricati.

La cartella "\BACKUP", invece, si suppone sia stata precedentemente creata con l'operazione di salvataggio configurazione descritta al §14.2, quindi è riservata al ripristino della configurazione completa (tutti i file) di un terminale in caso di sostituzione.

La selezione della cartella viene effettuata mediante la seguente schermata:

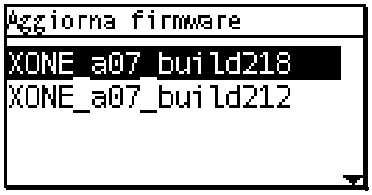


Usate i tasti freccia ▲▼ per selezionare una voce di menu e ↵ (Enter) per confermare.

Al termine si torna alla schermata "USB Menu".

14.4 AGGIORNAMENTO FIRMWARE DA CHIAVETTA USB

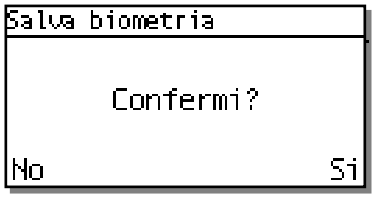
La quarta voce del menu "**Aggiorna firmware**", infine, consente l'aggiornamento in locale del firmware (utile per terminali *stand-alone* non collegati in Ethernet, o su cui comunque non sia possibile caricare il firmware via FTP): la chiavetta USB deve contenere nella *root* almeno un file chiamato "**XONE_***VNN***_build***nnn***.bin**" contenente la nuova versione di firmware (vedi §9 a pag.107). Se non viene trovato nessun file di questo tipo compare il messaggio "Operazione fallita", in caso contrario compare un'altra schermata dove vengono elencati tutti i file di questo tipo trovati (anche se ce n'è uno solo), ad esempio:



Usate i tasti freccia ▲▼ per selezionare quale versione di firmware caricare ↵ (Enter) per confermare. Anche per questa operazione compare l'ulteriore finestra di richiesta conferma già descritta in precedenza, ma al termine dell'operazione il terminale si riavvia automaticamente col nuovo firmware, uscendo quindi dal menu USB.

14.5 SALVATAGGIO DATI BIOMETRICI SU CHIAVETTA USB

Se X1/X2 è equipaggiato con un modulo biometrico esterno FingerBOX per la scansione di impronte digitali, e se la cartella locale **\BIOEXP** (normalmente non presente ma creata, assieme alla cartella **\BIOIMP** , subito dopo avere abilitato la gestione del modulo FingerBOX) contiene già almeno un file fra **BIOUPDATE.TXT** e **BIODATA.TXT** (vedi §11.1 a pag. 123), allora il menu USB include anche la voce "**Salva biometria**", visibile solo spostando la selezione verso il basso fino alla pagina successiva. Selezionando tale voce compare una richiesta di conferma:



Premete ↵ (Enter) (Si) per procedere e **[<-]->** (No) per abortire. A seguito di questa operazione, l'intero contenuto della cartella locale **\BIOEXP** verrà copiato in una cartella **\BIOEXP** che viene automaticamente creata nella *root* della chiavetta USB. Questa operazione viene di norma eseguita su un terminale "master", cioè quello dove vengono effettuate in locale tutte le operazioni di modifica di un archivio di impronte, con il quale si vogliono in seguito sincronizzare altri terminali X1/X2 "slave" facenti parte dello stesso impianto.

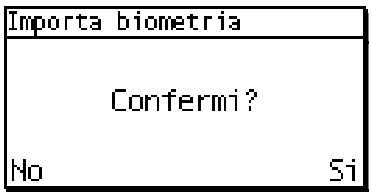
14.6 IMPORTAZIONE DATI BIOMETRICI DA CHIAVETTA USB

Se nella *root* della chiavetta USB inserita, è presente una cartella **\BIOEXP** che contiene almeno un file fra "**BIOUPDATE.TXT**", "**BIODATA.TXT** e "**FINGER**" (con o senza "**USERCODS**", vedi §11.3 a pag. 134), allora il menu USB include anche la voce "**Importa biometria**", visibile solo spostando la selezione verso il basso fino alla pagina successiva. Selezionando tale voce compare un'altra schermata che può contenere una, due o tre opzioni, a seconda di quali fra i file sopra menzionati siano presenti:

La voce "Resetta & importa" compare solo se è stato trovato un file **BIODATA.TXT**: selezionando questa opzione l'intero archivio di impronte del terminale sarà cancellato e verranno a quel punto importati tutti i dati biometrici contenuti nel file, come descritto al § 11.3 a pag. 134. La voce "Aggiorna archivio", invece, compare solo se è stato trovato un file **BIOUPDATE.TXT**: selezionando questa opzione l'archivio di impronte locale del terminale non viene cancellato, ma semplicemente aggiornato replicando tutte le operazioni elencate nel file, come descritto al medesimo paragrafo. Queste operazioni vengono di norma eseguite su un terminale "slave" che si vuole sincronizzare con il contenuto di un archivio di impronte creato su un terminale "master" facente parte dello stesso impianto, e sul quale si assume sia stata eseguita l'operazione di salvataggio dati biometrici

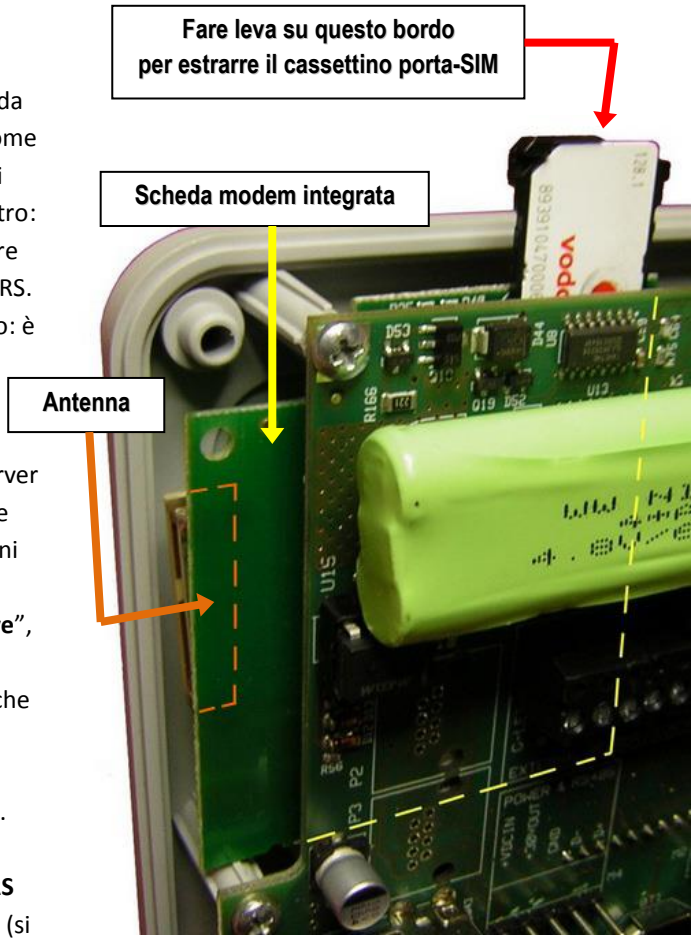
descritta al §14.5 qui sopra, usando la stessa chiavetta USB (questo è il motivo per cui X1/X2 cerca nella chiavetta la cartella **\BIOEXP** invece di una cartella "\BIOIMP" come ci si potrebbe aspettare per un'operazione di importazione). Infine, la voce "Aggiorna da FINGER" compare solo se è stato trovato un file **FINGER** generato in seguito all'esportazione dei dati biometrici contenuti su terminali 929 FingerTRAX+G/SU o 962 SuperTRAX con modulo biometrico esterno FingerBOX (lo stesso utilizzato da X1/X2): anche in questo caso l'archivio di impronte locale del terminale non viene cancellato, ma semplicemente aggiornato con i dati biometrici contenuti nel file FINGER e nell'eventuale file **USERCODS** associato (se anch'esso presente nella cartella **\BIOEXP**).

Tutte le opzioni chiedono conferma prima di procedere: premete ↵ (Enter) (Si) per procedere e **[<-]->** (No) per abortire.

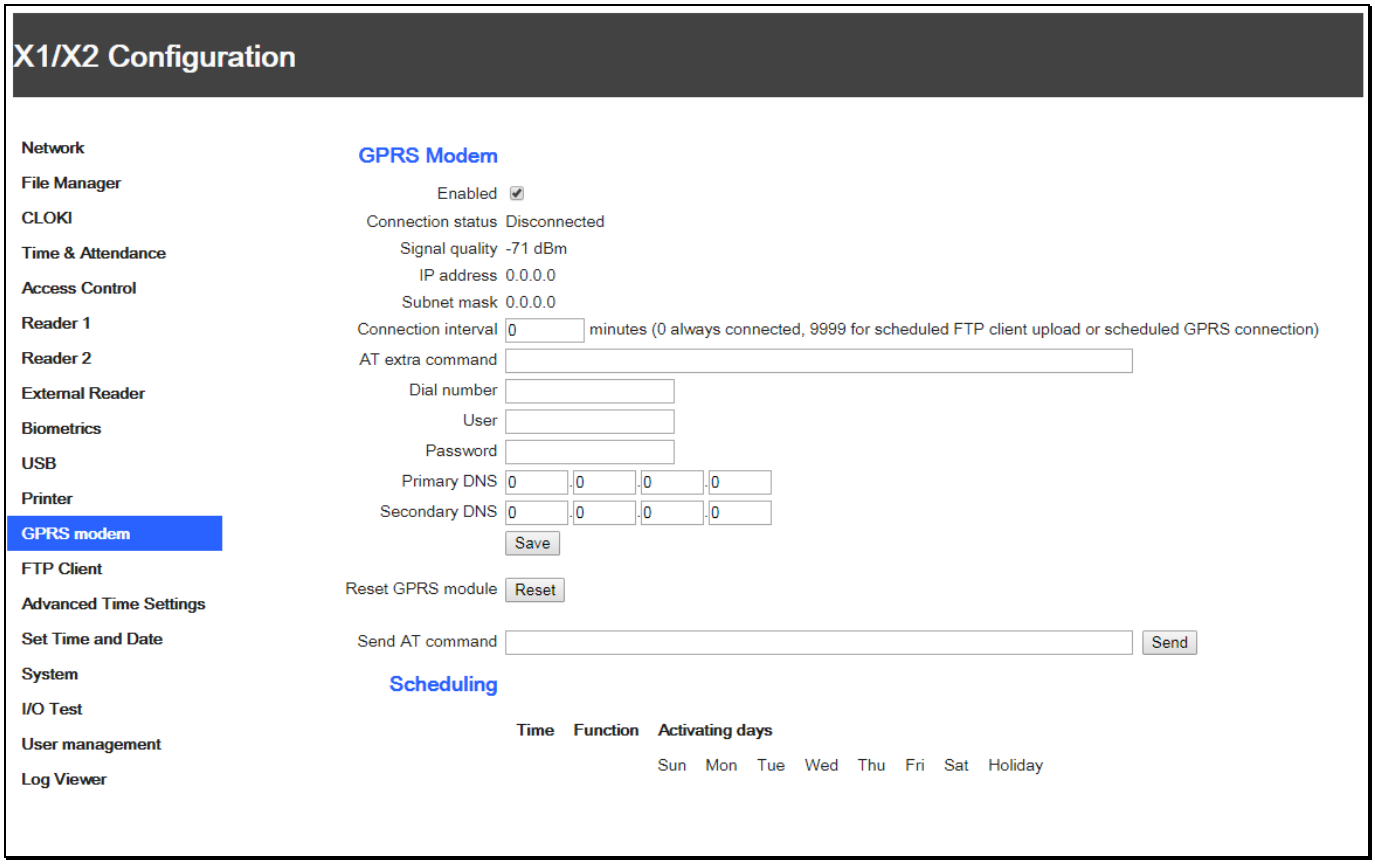


15. USO DEL MODEM GPRS OPZIONALE

X1/X2 è disponibile anche in versione con modem GPRS integrato (da richiedere espressamente all'atto dell'acquisto del terminale). La scheda modem viene montata in produzione sul lato nascosto della scheda principale, quindi non è visibile frontalmente ma solo lateralmente. In figura sono mostrate la posizione della scheda modem, dell'antenna e del cassettino estraibile dove inserire la scheda SIM (non in dotazione) necessaria per il funzionamento, a cui si può accedere dal lato superiore, subito sopra la batteria.



Il modem GPRS viene gestito tramite la stessa porta di comunicazione normalmente utilizzata per il lettore esterno da collegare sulla morsettiera a vite estraibile contrassegnata come "EXTERNAL READER". Non è quindi possibile usare entrambi i dispositivi allo stesso tempo, ma solo uno in alternativa all'altro: per evitare conflitti di natura elettrica si prega di non collegare un lettore sulla morsettiera a vite in presenza del modem GPRS. Anche se presente, per default il modem GPRS non è abilitato: è necessario attivarne la gestione impostando il parametro **Enabled**=1 nella sezione *[GPRS]* del file PARAMETERS.TXT (vedi §4.11 a pag. 57) oppure, analogamente, spuntando la *checkbox* "**Enabled**" nella pagina "**GPRS modem**" del web server HTTP (inizialmente questa è l'unica opzione disponibile in tale pagina: non appena spuntata, compaiono tutte le altre opzioni di configurazione della connessione GPRS che vedremo nel dettaglio più avanti). Una volta confermato col pulsante "**Save**", la pagina "**External reader**" del web server HTTP mostrerà la scritta rossa "**Reader used by GPRS modem**", per ricordarci che non sarà più possibile usare tale lettore (tuttavia è possibile continuare ad usare eventuali lettori aggiuntivi collegati su schede di espansione 914 NeoMAX opzionali, vedi §3.6 a pag. 11, le cui letture vengono comunque gestite secondo le impostazioni della pagina "**External reader**"). La pagina "**GPRS modem**" del web server HTTP ha l'aspetto mostrato in figura (si ricordi che è comunque possibile editare gli stessi parametri direttamente all'interno della sezione *[GPRS]* del file PARAMETERS.TXT, vedi §4.11 a pag. 57):



Oltre alla *checkbox* già spuntata per l'attivazione della gestione del modem GPRS (corrispondente al parametro **Enabled**) compaiono, nell'ordine:

- lo stato attuale della connessione GPRS
- la potenza del segnale rilevato al momento del caricamento della pagina, espressa in dBm.

  Nota 1: in assenza della scheda SIM, o comunque prima dell'effettiva registrazione alla rete GSM del fornitore di servizi scelto, il segnale rilevato potrebbe essere diverso da quello che sarà poi effettivamente utilizzato, e quindi anche la sua potenza.

  Nota 2: un segnale con potenza inferiore a -95dBm è in pratica troppo debole per consentire la comunicazione. Tenete presente che i rapporti fra le potenze espresse in dBm sono circa i seguenti: 3 dBm in più equivalgono ad un raddoppio, 10 dBm in più ad un aumento di 10 volte, 20 dBm in più ad un aumento di 100 volte.
- il campo di testo per impostare l'intervallo di tempo (in minuti) fra una connessione e quella successiva (corrispondente al parametro **ConnectionInterval**):
  - 0 (default): il modem rimane sempre collegato una volta effettuata la connessione GPRS al fornitore di servizi, almeno fino a quando non sarà quest'ultimo ad interromperla unilateralmente (in tal caso, comunque, il modem effettuerà subito un nuovo tentativo di connessione).
  - Ogni valore diverso da 0 o 9999: la prima connessione (che ha luogo non appena viene salvata la configurazione con il modem GPRS attivato e la definizione dei parametri necessari per l'accesso al servizio, o in seguito ad ogni successivo riavvio del terminale(*)) dura solo 5 minuti, trascorsi i quali viene valutato se vi sia in quel momento un'attività di comunicazione online significativa (X1/X2 gestito da Xatl@s, oppure tramite protocollo HTTP): in particolare, se gli ultimi 5 messaggi HTTP GET inviati dal terminale risultano essere solo di tipo "*Keep Alive*" (cioè non sono state trasmesse transazioni), e se le relative HTTP RESPONSE inviate dall'host contengono esclusivamente campi del tipo *screen*, *show*, *beep*, *time*, *date*, e *keepaliveperiod* (vedi §12.2 a pag. 139), allora la connessione GPRS viene interrotta, e tale rimane per un tempo pari al valore di **ConnectionInterval**. Allo scadere dell'intervallo viene effettuata una nuova connessione che dura solo 5 minuti, e così via.
  - 9999: il modem effettuerà la connessione GPRS solo se vi sono delle schedulazioni relative a connessioni GPRS e/o esportazioni da client FTP, da impostare mediante il file ALARMS.TXT (vedi §4.2 a pag. 18). In questo caso la connessione GPRS avviene solo in corrispondenza degli orari impostati, e viene chiusa automaticamente al termine di ciascuna esportazione da FTP client, o agli orari delle eventuali disconnessioni schedulate (se

specificate). In ogni caso, un host HTTP può forzare la chiusura connessione in qualunque momento usando il tag "**gprs=off**" nella risposta ai pacchetti "KeepAlive" ricevuti, vedi §12.4 a pag. 141.

Nota: le comunicazioni via protocollo FTP, e quindi sia i trasferimenti gestiti dal lato host collegandosi al server FTP del terminale, sia quelli gestiti autonomamente dal client FTP del terminale mediante esportazioni schedulate (vedi §7.3 a pag. 102), hanno luogo ad un livello superiore e indipendente da quello relativo alla gestione della connessione GPRS, il quale quindi non "vede" l'attività svolta via FTP. Per questo motivo, qualora si intenda usare il protocollo FTP, raccomandiamo di lasciare il parametro **ConnectionInterval** al valore di default (0), altrimenti la connessione GPRS potrebbe essere interrotta in qualunque momento, anche durante un trasferimento di file. In alternativa, ma solo per l'utilizzo del client FTP del terminale mediante esportazioni schedulate, è possibile impostare **ConnectionInterval** al valore "9999" (vedi sopra).

- il campo di testo per impostare il comando speciale per il modem GPRS che contiene il nome del punto di accesso alla rete (APN, *Access Point Name*), corrispondente al parametro **ATextraCommand**: si tratta di un parametro fondamentale per il funzionamento della connessione GPRS. Normalmente potete impostarlo al valore seguente:

  AT+CGDCONT=1,IP,<*APN*>**,,0,0**

  dove <*APN*> è una stringa contenente il nome del punto di accesso, che dipende dal fornitore di servizi scelto. Ad esempio, per l'Italia, per la rete Vodafone <*APN*>=**web.omnitel.it** mentre per la rete TIM <*APN*>=**ibox.tim.it**

  Nota: questo campo non deve contenere delle virgolette ""

- il campo di testo per impostare il numero telefonico da chiamare per collegarsi alla rete GPRS (corrispondente al parametro **Dialnum**): si tratta di un parametro fondamentale per il funzionamento della connessione GPRS. Normalmente potete impostarlo al valore ***99***1#**

  Se così non dovesse funzionare potete provare col valore ***99#**

- il campo di testo per impostare il nome utente per effettuare l'accesso alla rete GPRS (corrispondente al parametro **User**), solo se richiesto dal fornitore di servizi scelto

- il campo di testo per impostare la password per effettuare l'accesso alla rete GPRS (corrispondente al parametro **Password**), solo se richiesto dal fornitore di servizi scelto

- i campi di testo per impostare gli indirizzi dei server DNS pubblici primario e secondario che potrebbe essere necessario contattare per risolvere i nomi degli URL logici eventualmente usati per definire i parametri **MasterURL** e **ServerURL** (a seconda di quale dei due venga poi usato per comunicare), che si trovano rispettivamente nelle sezioni *[Ethernet]* e *[FtpClient]* del file PARAMETERS.TXT (vedi §4.11 a pag. 56 e 58). Se i suddetti parametri contengono già degli indirizzi IP pubblici e raggiungibili, l'impostazione dei campi DNS può essere omessa, altrimenti è necessaria poiché tali valori non vengono impostati automaticamente dal fornitore di servizi GPRS.

  Esempi di valori utilizzabili sono quelli dei DNS pubblici di Google: **8.8.8.8** e **8.8.4.4**

  Nota: questi campi in realtà corrispondono ai parametri **Primary_DNS** e **Secondary_DNS** che si trovano nella sezione *[Ethernet]* del file PARAMETERS.TXT, e che quindi vengono usati anche in caso di connessione Ethernet standard con DHCP disabilitato.

- il pulsante per effettuare un reset hardware del modem GPRS, da usare solo nel caso in cui non si riesca più a farlo funzionare correttamente

- il campo di testo per inviare un qualunque comando AT al modem GPRS. Dopo avere riempito il campo, per inviare effettivamente il comando premete il tasto **Send**: la risposta del modem verrà mostrata subito sotto al campo di testo una volta ricaricata la pagina web.

  Ad esempio, il comando **AT+CREG?** consente di verificare lo stato della registrazione della scheda SIM alla rete GSM. La corrispondente risposta del modem ha sempre il formato **+CREG: 0,***n* dove *n* può assumere i seguenti valori:

  0: Registrazione alla rete impossibile (controllare la potenza del segnale, lo stato della scheda SIM e l'eventuale richiesta di un codice PIN, vedi nota successiva)

  1: Registrazione effettuata nella rete domestica del fornitore della scheda SIM

  2: Ricerca rete (normalmente solo in fase di avvio)

  3: Registrazione vietata

  5: Registrazione effettuata in *roaming* (nella rete di un altro fornitore di servizi)

Nota: per un corretto funzionamento della connessione GPRS su X1/X2, la scheda SIM non deve richiedere l'inserimento di un codice PIN. Ricordatevi pertanto di inserire la scheda SIM in un telefono cellulare e disabilitare la richiesta del PIN, se necessario, prima di usarla con il modem GPRS. Il comando **AT+CPIN?** consente di verificare lo stato della richiesta PIN della scheda SIM. La corrispondente risposta del modem ha sempre il formato **+CPIN:** <*stato*> dove <*stato*> può assumere i seguenti valori:

SIM PIN: Richiesta PIN

SIM PUK: Scheda SIM bloccata, richiesta codice PUK (è stato inserito un PIN errato per tre volte consecutive)

READY: Scheda SIM pronta (richiesta PIN disabilitata, o inserito PIN corretto)

- le connessioni GPRS attualmente pianificate, come definito dal file ALARMS.TXT, se presente (vedi §4.2 a pag. 18)

(*) **Nota importante**: dopo un riavvio, il modem GPRS viene resettato e inizializzato, e dovrebbe registrarsi in breve tempo alla rete del provider (circa 25 secondi): affinché questo avvenga, deve essere disponibile una rete 2G per tale fornitore, e non deve funzionare in modalità di roaming. Se è stato impostato un **ConnectionInterval** diverso da 9999, circa 30 secondi dopo l'avvio il modem effettua automaticamente il primo tentativo di connessione GPRS. Se però è presente un cavo Ethernet collegato al connettore RJ45 di X1/X2, la connessione GPRS non viene mai effettuata, e nel caso in cui il cavo Ethernet venga collegato a connessione GPRS in corso, quest'ultima verrà subito interrotta. Se il cavo Ethernet viene rimosso, e sempre nel caso in cui sia stato impostato un **ConnectionInterval** diverso da 9999, entro 30 secondi il modem effettua automaticamente un tentativo di connessione GPRS.

Attenzione: ricordiamo che anche in caso di connessione di GPRS riuscita, se il parametro **MasterURL** nella sezione *[Ethernet]* del file PARAMETERS.TXT (vedi §4.11 a pag. 56) è stato impostato ad un valore non vuoto il terminale si aspetta di ricevere una risposta HTTP dal server online definito da tale parametro, altrimenti si disconnette automaticamente: questo accade dopo un tempo pari a circa una dozzina di volte il valore attuale del parametro **KeepAliveInterval** nella sezione *[Ethernet]*. Poi, entro 1 minuto, procederà con un nuovo tentativo di connessione GPRS e così via. Non appena il server risponde (questo significa che il terminale passa allo stato "server online"), la connessione diventa stabile, e dura la durata prevista.

15.1 VISUALIZZAZIONE STATO MODEM GPRS

Non appena viene attivata la gestione del modem GPRS secondo le modalità indicate nel paragrafo precedente, sul bordo superiore del display di X1/X2 compare un'icona relativa allo stato del modem GPRS:

Come si può vedere, tale icona si trova nella stessa posizione di quella relativa allo stato di carica della batteria, normalmente visualizzata in assenza di alimentazione (vedi §13 a pag. 147): in presenza di un modem GPRS attivato, pertanto, non è più possibile mostrare lo stato di carica durante il funzionamento a batteria.

Vediamo ora le possibili varianti di questa icona ed il loro significato:

 Segnale non misurabile o non ancora misurato

 Segnale con potenza minore o uguale a -95dBm

 Segnale con potenza compresa fra -93dBm e -73dBm

 Segnale con potenza maggiore o uguale a -71dBm

 Tentativo di connessione GPRS in corso

 Connessione GPRS stabilita

 Mancanza parametri di connessione fondamentali *oppure* modem bloccato (stato temporaneo)

16. ESECUZIONE DI COMANDI VIA FTP

E' possibile eseguire alcune operazioni da remoto usando un client FTP per caricare dei file con un nome specifico nella root del terminale: utilizzando uno dei nomi riservati elencati nel seguito, verrà eseguita l'operazione corrispondente ma non verrà in realtà creato alcun file con quel nome, per cui i file utilizzati a questo scopo possono anche essere vuoti.

Nota: per i nomi dei file si possono usare indifferentemente lettere maiuscole o minuscole.

- **$BIOEXP.CMD**

  Se X1/X2 è equipaggiato con un modulo biometrico esterno FingerBOX (e ne è stata abilitata la gestione), in seguito all'invio di questo file viene eseguita un'operazione di esportazione dei dati relativi a tutte le impronte attualmente presenti nel modulo (vedi anche §11.1 a pag. 130). Tale operazione può anche essere effettuata da remoto mediante il pulsante "**Export archive**" nella pagina "**Biometrics**" del web server HTTP del terminale, come visto al §11 a pag. 120.

- **$BIORESET.CMD**

  Se X1/X2 è equipaggiato con un modulo biometrico esterno FingerBOX (e ne è stata abilitata la gestione), in seguito all'invio di questo file viene eseguita un'operazione di cancellazione dell'intero contenuto dell'archivio di impronte (vedi anche §11.1 a pag. 130). Tale operazione può anche essere effettuata da remoto mediante il pulsante "**Delete all templates**" nella pagina "**Biometrics**" del web server HTTP del terminale, come visto al §11 a pag. 120.

- **$RECOVER.CMD**

  In seguito all'invio di questo file viene eseguita un'operazione di recupero di tutte le transazioni che sono state in precedenza registrate sul terminale, riesportando il contenuto di tutti i file **btransactions*.loc** ancora presenti in un apposito file TRANSACTION_BACKUP.TXT e nello stesso formato con cui vengono registrate nel file TRANSACTIONS.TXT. Tale operazione può anche essere effettuata da remoto mediante il pulsante "**Recover**" nella pagina "**System**" del web server HTTP del terminale, come visto al §7 a pag. 96.

- **$RECOVER***ggmmaaaa***.CMD**

Funziona come nel caso precedente ma consente di recuperare solo le transazioni registrate a partire dal giorno specificato: *gg*=giorno, *mm*=mese, *aaaa*=anno.

17. TRATTAMENTO SICURO DEI DATI AI FINI DEL GDPR

Con il regolamento generale sulla protezione dei dati (GDPR, General Data Protection Regulation - Regolamento UE 2016/679) la Commissione europea intende rafforzare e rendere più omogenea la protezione dei dati personali di cittadini dell'Unione Europea e dei residenti nell'Unione Europea, sia all'interno che all'esterno dei confini dell'Unione europea (UE). Il testo, pubblicato sulla Gazzetta Ufserverficiale Europea il 4 maggio 2016 ed entrato in vigore il 25 maggio dello stesso anno, inizierà ad avere efficacia il 25 maggio 2018.

Poiché i terminali AXESS TMC possono ricevere, trasmettere e memorizzare al loro interno alcuni dati sensibili che rientrano fra i dati personali oggetto del GDPR, abbiamo previsto alcune nuove misure di sicurezza (attivabili opzionalmente) che possono consentire, al responsabile del sistema di controllo accessi e/o rilevazione presenze, di ottenere la conformità a tale regolamento.

17.1 DATI PERSONALI SUI TERMINALI

17.1.1 FILE DI LOG

All'interno dei file di LOG possono essere contenuti dei dati personali in chiaro: ad esempio il messaggio di conferma mostrato a display dopo una transazione valida (che può contenere il nome dell'utente), il contenuto in chiaro dei record del file USERS che sono stati ricercati durante le transazioni (anche se tale file è stato caricato in formato criptato, vedi sotto), il contenuto dei record biometrici aggiunti da un server HTTP con un comando "BIOADD" (se inviati in formato non criptato).

Per default queste informazioni non sono comunque visibili: lo diventano solo nel caso in cui l'amministratore del sistema imposti il parametro **LogLevel** ai valori 0 oppure 1 (vedi §4.11 a pag. 51): in tal caso, il terminale registra gli eventi in maniera dettagliata (il che può implicare la presenza di dati personali in chiaro) per un tempo massimo di una settimana, dopodiché torna a registrare solo gli eventi principali, come se il parametro fosse tornato al valore 2. Solo se il terminale viene riavviato o viene effettuato un salvataggio dei parametri, il periodo di registrazione dettagliata viene prolungato di un'altra settimana.

Azioni da intraprendere

Avvisare l'amministratore del sistema che impostare il log ad un livello inferiore a 2 è consigliato solo per brevi periodi ed esclusivamente a scopo di debug, e può rendere visibili alcuni dati personali nel file system del terminale.

17.1.2 FILE USERS.TXT

Il file USERS.TXT contiene i nomi degli utenti associandoli ai codici tessera, e può essere creato da un server remoto (tramite un comando HTTP "RA" per ogni singolo record, o tramite l'invio del file completo, sempre via HTTP oppure via FTP) o direttamente tramite il web editor CLOKI (vedi §5.14 a pag. 84) già caricato sul terminale.

Siccome l'associazione "nome utente - codice tessera" è un dato personale e sensibile, è necessario proteggere tale file: questo si può fare utilizzando un apposito formato criptato (vedi §17.2 a pag. 159) al posto del formato standard in chiaro. Il file in formato criptato si deve chiamare **USERS.EF***x*, dove *x* identifica il tipo di cifratura (attualmente è supportato solo il valore '**1**', ovvero "**RC4 + codifica Base64**", pertanto viene gestito solo il file criptato **USERS.EF1**), e può essere correttamente interpretato solo sui terminali nei quali è stata impostata la relativa chiave di cifratura. All'interno dei record, il campo "identificatore univoco dell'utente" (chiave primaria, vedi §5.11 a pag. 77) rimane visibile in chiaro, mentre tutta la parte rimanente del record viene criptata.

Nota: nel caso in cui venga attivata l'indicizzazione, il file di indice per il file criptato **USERS.EF1** si chiamerà **USERS.ID1**, invece di USERS.IDX come nel caso standard, per non creare ambiguità.

Azioni da intraprendere

E' consigliabile attivare la crittografia sul terminale (vedi §17.2 a pag. 159): una volta fatto questo, nel caso in cui sia un server remoto a inviare i dati, esso deve inviare il file/i record in formato già criptato in base alla stessa chiave di cifratura impostata sul terminale: questo modo di procedere si è reso necessario per poter sopperire alla mancanza di protezione dovuta all'assenza di protocolli sicuri (HTTPS e FTPS) su alcuni terminali: X1/X2, AXGATE, AXDOOR. ***Nota***: il formato dei comandi HTTP resta sempre lo stesso: il cambiamento riguarda solo il contenuto del record inviato, che deve essere criptato o meno a seconda delle impostazioni del terminale. Se con la crittografia attivata viene inviato un file USERS.TXT in formato non criptato, esso verrà del tutto ignorato. Se invece viene inviato un record in formato non criptato all'interno di un file USERS.EF*x*, esso renderà di fatto inutilizzabile tutto il file.

Resta comunque possibile lasciare la crittografia disattivata e inviare il file/i record in chiaro, purché si prenda atto del possibile rischio per la sicurezza.

Al momento attuale, CLOKI non è ancora in grado di leggere e/o creare file in formato criptato.

17.1.3 DATI BIOMETRICI

Il terminale può ricevere dati biometrici da un server remoto, o generarli autonomamente tramite *enrollment* sul modulo biometrico di "console" FingerBOX. Come per il file USERS.TXT, il server remoto può inviare i singoli comandi HTTP "BIOADD" con i record biometrici o caricare direttamente un file completo BIOUPDATE.TXT o BIODATA.TXT, sempre via HTTP oppure via FTP. I comandi e i file vengono rimossi dopo che il loro contenuto è stato importato sul modulo biometrico Suprema: solo i file USERCODS.TXT e/o USERCODS_*NN*.TXT rimangono sul terminale, ma essi contengono solo dei codici tessera ed i relativi attributi funzionali, ma nessun dato biometrico né tantomeno i nomi degli utenti, quindi non creano un problema di privacy.

L'*enrollment* sul modulo biometrico di "console" FingerBOX genera, per ogni utente registrato, un record nel file BIOUPDATE.TXT, al fine di consentirne l'esportazione ad un altro terminale. Inoltre, il terminale può esportare con un'unica operazione tutti i dati biometrici contenuti all'interno del modulo biometrico di "console" FingerBOX, creando il file BIODATA.TXT che normalmente non è presente senza effettuare questa operazione.

Entrambi i file BIOUPDATE.TXT e BIODATA.TXT possono in generale contenere dati biometrici, quindi personali e sensibili. Ciascun *template* è costituito da una stringa di identificazione univoca in formato esadecimale, a partire dalla quale non è comunque possibile ricostruire l'immagine dell'impronta originale: nonostante questo, è teoricamente possibile appropriarsi dei dati biometrici presenti (anche solo temporaneamente) nel file system di un terminale e importarli su altri terminali senza autorizzazione, ossia in maniera fraudolenta. E' pertanto necessario proteggere tali file: questo si può fare utilizzando un apposito formato criptato (vedi il prossimo §17.2) al posto del formato standard in chiaro. I file in formato criptato si devono chiamare rispettivamente **BIOUPDATE.EF***x* e **BIODATA.EF***x*, dove *x* identifica il tipo di cifratura (attualmente è supportato solo il valore '**1**', ovvero "**RC4**", pertanto vengono gestiti solo i file criptati **BIOUPDATE.EF1** e **BIODATA.EF1**), e possono essere correttamente interpretati solo sui terminali nei quali è stata impostata la relativa chiave di cifratura. All'interno dei file e dei record inviati con i comandi HTTP "BIOADD" solo i *template* sono criptati, mentre tutto il resto rimane visibile in chiaro, inoltre ciascun *template* viene criptato singolarmente.

Azioni da intraprendere

E' consigliabile attivare la crittografia sul terminale (vedi il prossimo §17.2): una volta fatto questo, nel caso in cui sia un server remoto a inviare i dati, esso deve inviare il file/i record nei comandi HTTP "BIOADD" in formato già criptato in base alla chiave di cifratura impostata sul terminale: questo modo di procedere si è reso necessario per poter sopperire alla mancanza di protezione dovuta all'assenza di protocolli sicuri (HTTPS e FTPS) su alcuni terminali: X1/X2, AXGATE, AXDOOR. ***Nota***: il formato dei comandi HTTP resta sempre lo stesso: il cambiamento riguarda solo il contenuto del record inviato, che deve essere criptato o meno a seconda delle impostazioni del terminale. Se con la crittografia attivata viene caricato nella cartella **\BIOIMP** un file BIOUPDATE.TXT o BIODATA.TXT in formato non criptato, esso verrà del tutto ignorato; se invece viene inviato un comando HTTP "BIOADD" con un record in formato non criptato, quello specifico record non potrà funzionare.

Inoltre, se è stata attivata la crittografia, in seguito ad ogni *enrollment* il terminale aggiunge un record nel file BIOUPDATE.EF*x,* mentre in seguito ad una esportazione dei dati biometrici crea il file BIODATA.EF*x*: in tutti i casi, i record avranno i singoli template criptati.

Resta comunque possibile lasciare la crittografia disattivata e inviare il file/i record nei comandi HTTP "BIOADD" in chiaro, purché si prenda atto del possibile rischio per la sicurezza.

17.2 CRITTOGRAFIA SUI DATI PERSONALI

Quando si attiva la crittografia, attualmente il terminale utilizza esclusivamente l'algoritmo invertibile **RC4** (anche detto **Arcfour**), che è uno tra i più famosi e diffusi algoritmi di cifratura a flusso (*stream*) a chiave simmetrica (ovvero per il quale la chiave usata per criptare è la stessa usata poi per decriptare), utilizzato ampiamente in protocolli quali l'SSL ed il WEP. Necessita di una chiave di cifratura e genera un output binario di pari lunghezza rispetto all'input.

A seconda delle esigenze, all'output dell'algoritmo RC4 può anche essere applicato il sistema di codifica **Base64**, che consente di trasformarlo in una stringa ASCII di lunghezza pari a soli 2/3 della sequenza originaria in cifre esadecimali.

Poiché il terminale deve poter leggere la chiave di cifratura autonomamente in qualunque momento, è necessario memorizzarla da qualche parte: essendo ovviamente un rischio per la sicurezza scriverla in chiaro nel file system del terminale (memoria flash interna o micro-SD), si è deciso di salvarla nella memoria RAM tamponata dell'orologio del terminale: questo significa che se il terminale viene lasciato privo di alimentazione, quando la batteria a bottone al litio per il mantenimento dell'orologio (vedi §3.2 a pag. 8) si scarica, la chiave di cifratura viene persa rendendo inutilizzabili i file criptati attualmente presenti. E' comunque possibile reimpostare la stessa chiave di cifratura precedentemente memorizzata (se la si conosce) per ripristinare la piena funzionalià del terminale.

17.2.1 VERIFICA SUPPORTO CRITTOGRAFIA DA PARTE DEL TERMINALE

Un server remoto può verificare facilmente se un certo terminale dispone di un firmware che supporta la crittografia: è sufficiente usare il comando HTTP "**cmd=GETPAR [Info] CryptoSupport**". Quando si invia questo comando in risposta ad un "*Keep Alive*", il terminale risponde subito al server con un nuovo pacchetto HTTP GET "*Keep Alive*" a cui aggiunge in coda la stringa "**&parVal=Info,CryptoSupport**,*<valore>*&**cmd=ok**", dove il campo *<valore>* è uguale a "??" oppure "0" in caso di crittografia non supportata, e "1" in caso di crittografia supportata.

In generale, comunque, tutte le versioni di fw **g***NN***_build***nnn* e successive supportano la crittografia.

17.2.2 IMPOSTAZIONE DELLA CHIAVE DI CIFRATURA

L'impostazione della chiave di cifratura è l'operazione preliminare da effettuare, senza la quale non è neppure possibile attivare la crittografia, come vedremo più avanti. Anche se può essere scelta a piacere, affinché venga considerata valida la chiave di cifratura deve avere una lunghezza compresa fra 8 e 16 caratteri (lettere, numeri o simboli), e deve contenere almeno un numero, una lettera maiuscola ed una lettera minuscola: in caso contrario l'impostazione non avrà effetto. Per impostare la chiave è possibile usare uno qualunque dei seguenti metodi:

1) Impostazione del parametro temporaneo **CryptoKey** nella sezione *[System]* del file PARAMETERS.TXT mediante caricamento dell'intero file PARAMETERS.TXT; tale parametro viene letto e subito rimosso dal file PARAMETERS.TXT;
2) Impostazione del parametro temporaneo **CryptoKey** nella sezione *[System]* del file PARAMETERS.TXT mediante caricamento del file UPDATECONF.TXT con le sole righe necessarie (vedi anche §4.10 a pag. 28);

   [System]
   CryptoKey=*<valore>*

   Tale parametro viene letto ma non viene comunque aggiunto al contenuto del file PARAMETERS.TXT, inoltre il file UPDATECONF.TXT viene immediatamente cancellato;

3) Impostazione del parametro criptato temporaneo **CryptoKey** nella sezione *[System]* del file PARAMETERS.TXT mediante il comando HTTP "**cmd=SETPAR [System]**<TAB><TAB>**CryptoKey=**<*valore*>"; tale parametro viene letto ma non viene comunque aggiunto al contenuto del file PARAMETERS.TXT;
4) Impostazione diretta della chiave di cifratura mediante la sezione "**Cryptography password**" della pagina "**System**" del web server http del terminale.

Nei casi 1), 2) e 3) citati qui sopra, per ovvi motivi di sicurezza il valore specificato per il parametro **CryptoKey** non deve essere la chiave di cifratura in chiaro, bensì la stringa ottenuta criptando la chiave stessa a sua volta con l'algoritmo **RC4 + codifica Base64**, in questo caso però usando come ulteriore chiave l'indirizzo MAC esteso del terminale (solo le cifre senza i caratteri di separazione): infatti questa ulteriore chiave deve essere nota al terminale in modo che esso possa decrittare la stringa ricevuta ed ottenere così la vera chiave di cifratura. ***Attenzione***: per questa applicazione dell'algoritmo **RC4**, occorre ricordarsi di specificare che la chiave MAC deve essere inserita come dato di 8 bytes in formato esadecimale, invece che come stringa di testo di 16 caratteri ASCII. Naturalmente questo significa anche che è relativamente semplice risalire alla vera chiave di cifratura se si riesce ad intercettarne il valore criptato: per questo motivo i metodi 1), 2) e 3) sono sconsigliati se la comunicazione con il server remoto avviene in assenza di protocolli sicuri (HTTPS e FTPS).

Esempio:

Chiave di cifratura: ***Password#123***

MAC del terminale: ***000424B47620E9C2***

Applicazione dell'algoritmo RC4 usando come chiave hex il MAC: ***7A7BE7630068B675B07F9EB8***

Applicazione della codifica Base64: ***envnYwBotnWwf564***

Valore da utilizzare: ***CryptoKey=envnYwBotnWwf564***

Nel caso 4) la chiave di cifratura viene inserita in modo mascherato direttamente dall'interfaccia del browser utilizzato, via HTTP: anche in questo caso, quindi, si tratta di un tipo di comunicazione non sicuro.

In tutti i casi, se la chiave inserita soddisfa i criteri di validità, nella sezione "**Cryptography password**" della pagina "**System**" del web server http del terminale deve comparire lo stato "**Saved**", in caso contrario deve comparire "**Invalid**" (come per default).

17.2.3 VERIFICA CORRISPONDENZA CHIAVI DI CIFRATURA IMPOSTATE SU TERMINALE E SERVER

Affinché il server remoto possa inviare il file/i record già nel corretto formato criptato corrispondente alla chiave di cifratura impostata sul terminale, è necessario essere sicuri che essa coincida con quella impostata sul server stesso. A tale scopo è possibile inviare al terminale l'apposito comando http "**GETPAR [Info] CryptoHash**", che per ovvi motivi di sicurezza non restituisce la chiave di cifratura in chiaro, bensì il relativo valore di *hash* calcolato mediante un algoritmo non invertibile (**PBKDF2**) a cui si applica come chiave ("*salt data*") la chiave di cifratura stessa, con 512 iterazioni e per una lunghezza di 16 bytes (32 cifre esadecimali). Quando si invia questo comando in risposta ad un "*Keep Alive*", il terminale risponde subito al server con un nuovo pacchetto HTTP GET "*Keep Alive*" a cui aggiunge in coda la stringa "**&parVal=Info,CryptoHash**,<*valore*>**&cmd=ok**", dove il campo <*valore*> corrisponde al valore di *hash* della chiave di cifratura calcolato come descritto sopra. Il server remoto deve quindi applicare lo stesso algoritmo alla chiave di cifratura in suo possesso per poter confrontare fra loro i due valori di *hash*: se sono identici, ne consegue che anche le rispettive chiavi di cifratura impostate lo sono.

Esempio:

Chiave di cifratura: ***Password#123***

Salt data: ***Password#123***

Iterazioni: ***512***

Lunghezza in byte: ***16***

Hash esadecimale in base all'algoritmo PBKDF2: ***639C4FE03AFBD3A2849BDA331A81EA32***

Campo contenuto nella HTTP GET "Keep Alive" inviata dopo la ricezione del comando:
&parVal=Info,CryptoHash,639C4FE03AFBD3A2849BDA331A81EA32&cmd=ok

17.2.4 ATTIVAZIONE DELLA CRITTOGRAFIA

Solo nel caso in cui sia già stata impostata una chiave di cifratura valida (vedi §17.2.1 a pag. 159), è quindi possibile attivare la gestione della crittografia sul terminale. Questo può essere fatto impostando il parametro **CryptoEnabled**=1 nella sezione *[System]* del file PARAMETERS.TXT (vedi §4.11 a pag. 53): in tal caso, se non è stata impostata una chiave di cifratura valida, il parametro torna immediatamente al suo valore di default (0). La stessa cosa si può fare spuntando la checkbox "**Encryption Enabled**" nella pagina "**System**" del web server http: in questo caso, se non è stata impostata una chiave di cifratura valida, la checkbox rimane ingrigita e non è proprio modificabile.

17.2.5 FORMATO DEI FILE CRIPTATI

Per quanto riguarda il file criptato **USERS.EF***x*, all'interno dei record il campo "identificatore univoco dell'utente" (chiave primaria, vedi §5.9 a pag. 77) rimane visibile in chiaro, mentre tutta la parte rimanente del record viene criptata: il dato da cui partire va inserito come stringa di testo ASCII, e all'output binario dell'algoritmo RC4 viene poi applicato il sistema di codifica **Base64** che consente di trasformarlo in una stringa ASCII, in modo tale da mantenere il file nel suo consueto formato testuale, per facilitarne la ricerca e l'indicizzazione. La cifratura **RC4 + codifica Base64** corrisponde al valore *x*='**1**' nell'estensione del file **USERS.EF1** (unico valore attualmente supportato).

Nota: nel caso in cui venga attivata l'indicizzazione, il file di indice per il file criptato **USERS.EF1** si chiamerà **USERS.ID1**, invece di USERS.IDX come nel caso standard, per non creare ambiguità.

Esempio:

Chiave di cifratura: ***Password#123***

Record di USERS.TXT non criptato:

0000000002_1234_John Snow _1801010000_1812312359_1_1_0000000000

Parte da criptare:

1234_John Snow _1801010000_1812312359_1_1_0000000000

Applicazione dell'algoritmo RC4 usando la chiave di cifratura ASCII:

0CF8F0D01408D3B0153DF9B0147CED69A98F1A0D0F3506853C3D18E103D75422354F42DF15C447DEC219D144B60B5E29F2D2181604218B7E95E71D719B84

Applicazione della codifica Base64:

DPjw0BQI07AVPfmwFHztaamPGg0PNQaFPD0Y4QPXVCI1T0LfFcRH3sIZ0US2C14p8tIYFgQhi36V5x1xm4Q=

Record di USERS.EF1 criptato:

0000000002_DPjw0BQI07AVPfmwFHztaamPGg0PNQaFPD0Y4QPXVCI1T0LfFcRH3sIZ0US2C14p8tIYFgQhi36V5x1xm4Q=

Per quanto riguarda invece i file criptati **BIOUPDATE.EF***x* e **BIODATA.EF***x*, all'interno dei file e dei record inviati con i comandi HTTP "BIOADD" solo i *template* sono criptati, mentre tutto il resto rimane visibile in chiaro. Ciascun *template* viene criptato singolarmente inserendolo come dato di partenza di 384 byte in formato esadecimale, invece che come stringa di testo di 768 caratteri ASCII: non c'è bisogno di conversioni poiché in realtà esso è già costuituito da una sequenza di cifre esadecimali che sono state scritte nel file usando un carattere ASCII per ogni cifra; analogamente, l'output dell'algoritmo RC4 non viene in questo caso ulteriormente convertito, ma solo letto in formato esadecimale, semplicemente scrivendo un carattere ASCII per ogni cifra, per un totale di 768 caratteri. Nei record che contengono 2 *template*, le relative versioni già criptate vengono poi concatenate fra loro: in tutti i casi si ottiene così un record avente la stessa lunghezza di quello senza cifratura. La cifratura **RC4** corrisponde al valore *x*='**1**' nell'estensione dei file **BIOUPDATE.EF1** e **BIODATA.EF1** (unico valore attualmente supportato).

Esempio:

Chiave di cifratura: ***Password#123***

Record di BIOUPDATE.TXT / BIODATA.TXT non criptato:

0000004465138711_20180612_02_0384_452410168B2A55462E01414B881903B100882983B14589340411A1841545B00D9107C6D01B0A1487501A1328C7D19B012487E0A0892608203F850FC850768B338880978312C8F1719107899022 0623C9909D85138A00331220CA00A68B1A8A41336C0F4A502B8A1C4A504142158AD0328F144AE1870C08CB702B8916 CC408C8809CDA0340D37CE009C841B4EA040883AD080A9100F512093072951A0A58F33D1A0AA8F1192F0A0891F5320 A48A319340AF8D31942004862495008108FFFF0001112233FFFFFE00011222334FFFFEE00112233344FFDDE0011233 3444FCCDDE0123334444FBCCDE0123444444BBBBCD0234444554AAAABC0234555555AAAAAC0234555554A9999A0344 555554999888644555555598877765555555558877666555554445877666554455444476665544444444476665554

44443333F555554444333223FF44444443322211FF44333332221100FFF333332111000FFFFFF222110000FFFFFFFF
F111000FFF0000000000000000000000000000000000000045251015892A55461A01B003092B82314508358291A0
0516C3E00D9109C4E0190A160580181424C5F0A28F2986119B08110670750A2746703C873446F0970313C731719208
C790210915C851289721C860A58C1048A02B8A1BC8A038691D48B03FBE154911828E27C9109883168930308D09C990
2B8A0609D0260B170A308A880A8BD0328D37CCA09B051B8D003F890D4E2092093A0F20A692291010A48F329040A490
1111009F0A1E1160A10B3011D0AD0D3012B000081C5351A80C23D351AD0BFFFEE0011223334FFFCDE00112333444FC
CDDE0122334444FCCCDE0123444445BBBBCD0124444455ABBBBCE134555555AAAAACE234555555AAA9ABE234555555
A99999744455555599887775555555559887776555555555887766655555555577766655455544447666655544444
4466555555444433336554454444333223544444443322211F43333332111100FFF333222111000FFFFF222211000
0FFFFFFFF1100000FFFF000000000000000000000000000000000000000000000000000_0_0_2_0_0_0_00_0

Applicazione dell'algoritmo RC4 usando la chiave di cifratura ASCII sul primo template (in rosso):

78EED3F2C068E99E551CEB95F312CEF8892713AE9E50AF911873885D26A3D41E9478B43F51FF6B68A030F35E42EFFC
77E70AC9E9BD37B36E9A522289FBC2E200BEB265A107DF1E8C694B23DD2C87EDD2703605CF1E5746540F3C62ACE179
33B8BF0B1CD4D4C338E0649C9FCD468A0603E8A6D707E8F9A5FC4FBA47E9EF699CCEA702E4FCA5803FE3B95B08F24F
FE0B16D8EA972F9F919841555AC00952875825731B40AFFD0D7ECEDB1A2ADFED60002757656396F527382929FFAF4E
1D8D1731171476C6C3F903FF3EE9C30FF060194B11071945FDC1F1BF1E1B8E2EBD75B8B42634011681A7CC681D640A
B601556AC0A37DE050A7FF914E00D580721765D6632FACD8509F2F367242450B1BA0103BA742CB1B22D235387C9306
DB7EE2350BE037CCB67620B17178C496F17D010D47AC06952C458DB01BE4CCA790F79E9064AC45CF3AAA6E30D7851E
F5CABF10892CBF800BD994B0C654B3D460DCD83E9DC8863776134BFBEA2D84819C1B7AD2831EB53434337805F47E56
FAC42422B0087A06

Applicazione dell'algoritmo RC4 usando la chiave di cifratura ASCII sul secondo template (in verde):

78EFD3F1C268E99E611C1ADD72204F78CCA70FAFBEB523B3DF8224483A22840A0F69776F52E15B2A00886F5F032FFC
7ED28B593C3E36FD3E995019075B236A20F10383B01DD07FDCF159621C2616EFD3800E0CCCDCF75ECC343EC232037E
F149B3D906D7956A3CDBE7DC460C4589E601EABB5476447FAB3EEFB7C5E829199ACEA44494FA25803D43BEDA0831EF
810A21464AAC36AACF9874C05A81B95387596493155FAF7EFD414DDA586ADA6C6182B755E562106523B611EFAEB64A
C1A1469D0DC9BBD93D1520CD63343DA00E8CCA7E47D8E7DF033E22FA48D476B4409B22F140BCFE9F7B4F562D6B8BF5
4E5BDAF297C2930FA8EF617819612A7C8C5FFB2F344E50149D42F2F9245389D6384CDDE6965206C9002FE71A4C93DB
F94D0F1738D037EE8455F2824269C4B5D35C232F56BC06B73D54AE920A83BB86F7812F8602CA22A84CCDD547B1E04C
33DB9DDCEC0E9CBDD6FBA66D2477800A8E2EF5E0723AABD989EC451415D28A6F9C1485D2831EB53434337805F47E56
FAC42422B0087A06

Record di BIOUPDATE.EF1 / BIODATA.EF1 criptato:

0000004465138711_20180612_02_0384_78EED3F2C068E99E551CEB95F312CEF8892713AE9E50AF911873885D26A3
D41E9478B43F51FF6B68A030F35E42EFFC77E70AC9E9BD37B36E9A522289FBC2E200BEB265A107DF1E8C694B23DD2C
87EDD2703605CF1E5746540F3C62ACE17933B8BF0B1CD4D4C338E0649C9FCD468A0603E8A6D707E8F9A5FC4FBA47E9
EF699CCEA702E4FCA5803FE3B95B08F24FFE0B16D8EA972F9F919841555AC00952875825731B40AFFD0D7ECEDB1A2A
DFED60002757656396F527382929FFAF4E1D8D1731171476C6C3F903FF3EE9C30FF060194B11071945FDC1F1BF1E1B
8E2EBD75B8B42634011681A7CC681D640AB601556AC0A37DE050A7FF914E00D580721765D6632FACD8509F2F367242
450B1BA0103BA742CB1B22D235387C9306DB7EE2350BE037CCB67620B17178C496F17D010D47AC06952C458DB01BE4
CCA790F79E9064AC45CF3AAA6E30D7851EF5CABF10892CBF800BD994B0C654B3D460DCD83E9DC8863776134BFBEA2D
84819C1B7AD2831EB53434337805F47E56FAC42422B0087A0678EFD3F1C268E99E611C1ADD72204F78CCA70FAFBEB5
23B3DF8224483A22840A0F69776F52E15B2A00886F5F032FFC7ED28B593C3E36FD3E995019075B236A20F10383B01D
D07FDCF159621C2616EFD3800E0CCCDCF75ECC343EC232037EF149B3D906D7956A3CDBE7DC460C4589E601EABB5476
447FAB3EEFB7C5E829199ACEA44494FA25803D43BEDA0831EF810A21464AAC36AACF9874C05A81B95387596493155F
AF7EFD414DDA586ADA6C6182B755E562106523B611EFAEB64AC1A1469D0DC9BBD93D1520CD63343DA00E8CCA7E47D8
E7DF033E22FA48D476B4409B22F140BCFE9F7B4F562D6B8BF54E5BDAF297C2930FA8EF617819612A7C8C5FFB2F344E
50149D42F2F9245389D6384CDDE6965206C9002FE71A4C93DBF94D0F1738D037EE8455F2824269C4B5D35C232F56BC
06B73D54AE920A83BB86F7812F8602CA22A84CCDD547B1E04C33DB9DDCEC0E9CBDD6FBA66D2477800A8E2EF5E0723A
ABD989EC451415D28A6F9C1485D2831EB53434337805F47E56FAC42422B0087A06_0_0_2_0_0_0_00_1

17.3 HASH SU UTENZE E PASSWORD

Per maggiore sicurezza, è ora possibile cambiare sia il nome utente che la password per ciascuno dei 2 possibili account usati per l'accesso remoto ai server HTTP/FTP del terminale: l'account amministratore (default: "admin" sia come nome utente che come password), che consente la gestione completa del terminale, ossia la configurazione avanzata sia via web server che via FTP (incluse tutte le operazioni potenzialmente "pericolose", come cambiare i parametri di configurazione, cancellare file, formattare la memoria, ecc.), e l'account per il solo accesso al web editor integrato CLOKI (vedi §5.14 a pag. 84, default: "manager" sia come nome utente che come password), che consente solo di modificare i file di servizio personalizzati (ad esempio le tabelle di controllo accessi, gli eventi temporizzati, i messaggi personali) e di visualizzare le timbrature effettuate. I nomi utente e le password per l'accesso remoto possono essere cambiati in modo indipendente mediante la pagina "**User Management**" del web server http.

Inoltre, sia i nomi utente e le password per l'accesso remoto che le password per l'accesso al menu supervisore dalla console del terminale (vedi §10.5 a pag. 113, default: "00000") e per l'accesso al menu USB all'inserimento di una chiavetta sul terminale (vedi §14 a pag. 148, default: "00000") vengono salvati nel file PARAMETERS.TXT, ma non in chiaro: al loro posto vengono salvati i relativi valori di *hash* calcolati mediante un algoritmo non invertibile (**PBKDF2**) a cui si applica una chiave ("*salt data*") fissa e riservata, con 1024 iterazioni e per un output di lunghezza pari a 32 bytes (64 cifre esadecimali), si vedano a tale proposito i parametri **SecureOperatorPassword**, **RemoteUsernameCrypto**, **SecureRemotePassword**, **ManagerUsernameCrypto**, **SecureManagerPassword** nella sezione *[System]* del file PARAMETERS.TXT (vedi §4.11 a pag. 52), e il parametro **SecurePasswordUSB** nella sezione *[USB]* dello stesso file (vedi §4.11 a pag. 59).

17.4 HASH SU TRANSAZIONI ED EVENTI

I dati relativi alle transazioni ed alle emissioni / rientri di eventi non hanno bisogno di essere nascosti, poiché contengono solo codici e quindi non stabiliscono un'associazione univoca con gli utenti. Tuttavia, per evitare modifiche fraudolente alle transazioni / eventi registrati, è possibile aggiungere un campo "valore di *hash*" a ciascun record nei file TRANSACTIONS.TXT (vedi §7 a pag. 96) e btransaction.loc, calcolato a partire dai dati essenziali di ciascuna transazione / evento: codice personale su 20 cifre allineato a sinistra con riempimento di zeri a destra (nel caso degli eventi tali 20 posizioni sono tutte riempite con dei caratteri spazio " "), data su 6 cifre in formato *AAAAMMGG*, ora su 6 cifre in formato *hhmmss*, direzione di transito su 1 cifra ('0'=uscita, '1'=entrata), indirizzo MAC esteso del terminale su 16 cifre esadecimali, tutti concatenati fra loro senza caratteri separatori) mediante un algoritmo non invertibile (**PBKDF2**) e usando come chiave ("*salt data*") la stessa chiave di cifratura utilizzata anche per la crittografia dei dati personali degli utenti (vedi §17.2.1 a pag. 159), con 512 iterazioni e per un output di lunghezza pari a 16 bytes (32 cifre esadecimali). Questo consente un controllo di congruenza delle transazioni / eventi dopo averli scaricati: il server remoto deve quindi applicare lo stesso algoritmo ai dati essenziali di ciascuna transazione / evento (codice personale, data e ora concatenati fra loro) per poter confrontare fra loro i due valori di *hash*: se sono identici, ne consegue che la transazione /evento è regolare.

17.4.1 ATTIVAZIONE DELL'HASH SU TRANSAZIONI ED EVENTI

Solo nel caso in cui sia già stata inserita una chiave di cifratura valida (vedi §17.2.1 a pag. 159), è quindi possibile attivare l'aggiunta del campo "valore di *hash*" alle transazioni / eventi. Questo può essere fatto impostando il parametro **TrnsHash**=1 nella sezione *[System]* del file PARAMETERS.TXT (vedi §4.11 a pag. 53): in tal caso, se non è stata impostata una chiave di cifratura valida, il parametro torna immediatamente suo al valore di default (0). La stessa cosa si può fare spuntando la checkbox "**Transaction hash**" nella pagina "**System**" del web server http: in questo caso, se non è stata impostata una chiave di cifratura valida, la checkbox rimane ingrigita e non è proprio modificabile.

Avvertenza: se il file btransaction.loc corrente è stato creato con una versione di fw precedente alla **g02_build***nnn* e vi sono timbrature ancora "pendenti" (cioè non ancora ricevute e/o non confermate da un server HTTP), entrambi i file continuano ad essere registrati nel formato standard: solo quando il server avrà ricevuto (e confermato) tutte le transazioni / eventi già presenti nel file btransaction.loc, oppure in seguito ad una cancellazione manuale o alla rinomina automatica di tale file al verificarsi di una delle condizioni descritte in dettaglio al §7 a pag. 96, il file btransactions.loc verrà ricreato e le transazioni / eventi inizieranno ad essere registrate in entrambi i file con l'aggiunta del campo "valore di *hash*".

Nota: nel caso in cui si sia definito un formato personalizzato per il file TRANSACTIONS.TXT (vedi §7.2 a pag. 101), per inserire il campo "valore di *hash*" (in qualunque posizione si desideri), è necessario usare il segnaposto '**Z**'.

Esempio:

Indirizzo MAC esteso del terminale: ***00:04:24:B6:96:56:C1:26***

Badge ***1944244558*** *letto in direzione* ***Entrata*** *alle* ***13:13:54*** *in data* ***12/07/2018***

Transazione in formato standard: ***20180712,131354,1,,1944244558,00,0,1,,,,,,***

Dato di cui fare l'hash: ***19442445580000000000201807121313541000424B69656C126***

Salt data (chiave di cifratura): ***Password#123***

Iterazioni: ***512***

Lunghezza in byte: ***16***

Hash esadecimale in base all'algoritmo PBKDF2: ***F4BB1A9235378D1734CECD819B3658E4***

Transazione in formato con hash: ***20180712,131354,1,,1944244558,00,0,1,,,,,,,F4BB1A9235378D1734CECD819B3658E4***

Transazione in formato standard con payload abilitato: ***20180712,131354,1,,1944244558,00,0,1,,,,,,,|00061944244558***

Transazione in formato con hash e con payload abilitato:
20180712,131354,1,,1944244558,00,0,1,,,,,,,F4BB1A9235378D1734CECD819B3658E4|00061944244558

Esempio di formato personalizzato con hash: ***CCCCCCCCCC-hhmmss-DDMMYYYY-V-Z***

Transazione in formato personalizzato con hash: ***1944244558-131354-12072018-1-F4BB1A9235378D1734CECD819B3658E4***

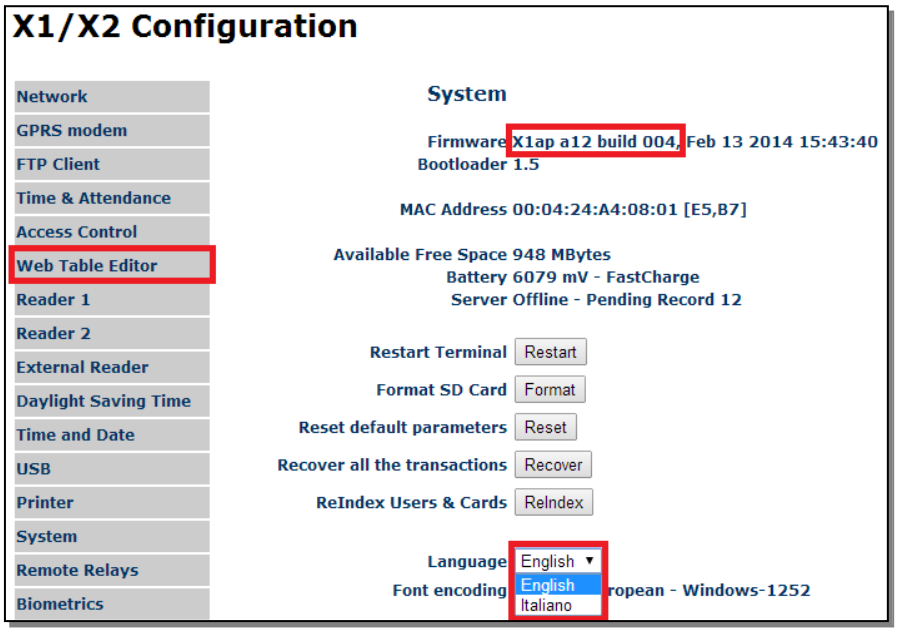
18. LA VERSIONE SPECIALE PER SERRATURE WIRELESS OFFLINE DELLA SERIE APERIO

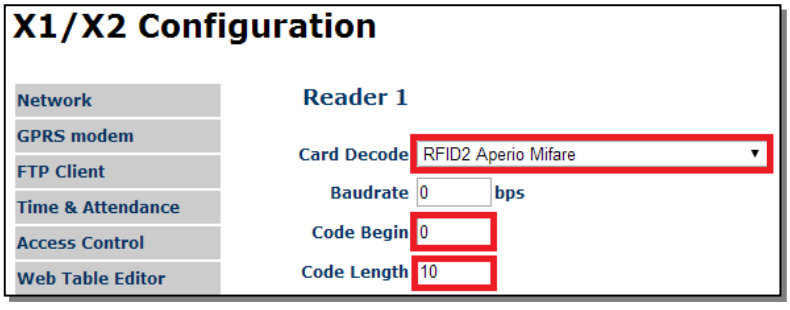
***Nota importante**: questo capitolo è momentaneamente disponibile solo in lingua inglese, verrà tradotto nelle successive versioni.*

A X1/X2 terminal with embedded 13,56MHz Mifare R&W module (p/n **930.0***x***0.***x***4**) can be provided with a special firmware version called "**XONE_ap_***VNN***_build***nnn*.bin**", which allows to use it as an "Aperio Plant Manager". In this mode, you can fully manage an Aperio series offline wireless locks plant, i.e. a plant with wireless locks working in stand-alone mode: compared to an online wireless lock plant (where each lock simply reads a card code, transmits it to a hub in real time and then waits for a reply), in this case each wireless lock not only reads the card code, but also checks the access authorizations in some key-protected Mifare sector of the so-called ACCESS CARDS, autonomously validates the access attempt basing upon this information and the current lock configuration, stores in other key-protected sectors a record including the current time (basing upon the wireless lock's internal clock setting) and the access attempt result, and finally unlocks the door in case of valid access.

Notes: 1) The special "X1/X2 APERIO ENCODER" FW activation key (see §4.12 at page 65) must also be introduced in order to use the X1/X2 as an "Aperio Plant Encoder", i.e. to allow the ACCESS CARDS creation starting from empty Mifare cards (all other functions are supported anyway, including locks authorizations changes on already formatted ACCESS CARDS); 2) due to memory space reasons, the special Aperio firmware version only provides two available languages for the on-screen user interface: English and Italian; 2) the following is valid only for firmware versions **XONE_ap_a12_build454** or subsequent.

In order to be able to manage the LOCK SETUP CARDS and the ACCESS CARDS of an offline Aperio wireless locks plant, it's also necessary to have the optional "access control tables web editor" firmware extension loaded (see §5.14 at page 84). So, first of all, make sure that the firmware is suitable and up-to-date, and that the optional ACTABLES folder is loaded and the "**Web Table Editor**" link in the HTTP web server pages is functional when clicked (check the parts highlighted in red in the picture beside). ***Note***: the following is valid for Web Table Editor versions **1.12** or subsequent.





Once done that, make sure that the **Card Decode** setting in the "**Reader 1**" page of the HTTP web server is set to "RFID2 Aperio Mifare", and that **Code Begin**=0 and **Code Length**=10 (check the parts highlighted in red in the picture beside): those are automatically set as the default values once the special Aperio firmware version has been loaded, and any change attempt won't be applied.

In the following, a X1/X2 terminal configured as above will be referred to as "X1/X2 Aperio Plant Manager".

18.1 X1/X2 APERIO PLANT MANAGER BASIC PLANT CONFIGURATION

In order to correctly configure X1/X2 Aperio Plant Manager for the management of your specific offline Aperio wireless locks plant, you need to contact Zucchetti Axess (should you have already installed other plants, you must also specify the current plant's facility code, which is a 4-digit progressive number starting from "0001"for the first plant installed, "0002" for the second, and so on). We will then provide you with the encrypted Mifare configuration string relevant to your specific offline Aperio wireless locks plant, which can be entered either by means of the "**Command**" box available in the "**Reader 1**" page of the HTTP web server, or by uploading a file named **READER1.TXT** (with a single line containing the encrypted Mifare configuration string) to X1/X2 Aperio Plant Manager and then changing any parameter (or simply restarting the terminal): that will configure the 13,56MHz R&W module with the proper Mifare authentication keys, and also automatically update all parameters in the *[Aperio]* section.

We recommend to save the encrypted Mifare configuration string of each specific offline Aperio wireless locks plant in a safe place: in case afterwards the *[Aperio]* section of the PARAMETERS.TXT file (see §4.11 at page 60) on a X1/X2 Aperio Plant Manager happens to be not properly configured (remember that all parameters in this section must not be manually edited or copied from a terminal to another), or if the embedded 13,56MHz Mifare R&W module for some reasons has a wrong configuration, you'll be able to enter its specific encrypted Mifare configuration string again as described above.

Note: X1/X2 Aperio Plant Manager can only manage plants for which has been defined 1 "standard sector"(*) plus a maximum of 8 total "alarms" + "access matrix" sectors(*) on each ACCESS CARD: remember that this information is included in the relevant encrypted Mifare configuration string. For instance, with 7 alarms sectors and 1 access matrix sector, each ACCESS CARD may store up to 7 x 6 = 42 events (typically valid / not valid access attempts on any wireless locks and, in rare cases, internal lock events such as a battery fault) and up to 96 lock authorizations.

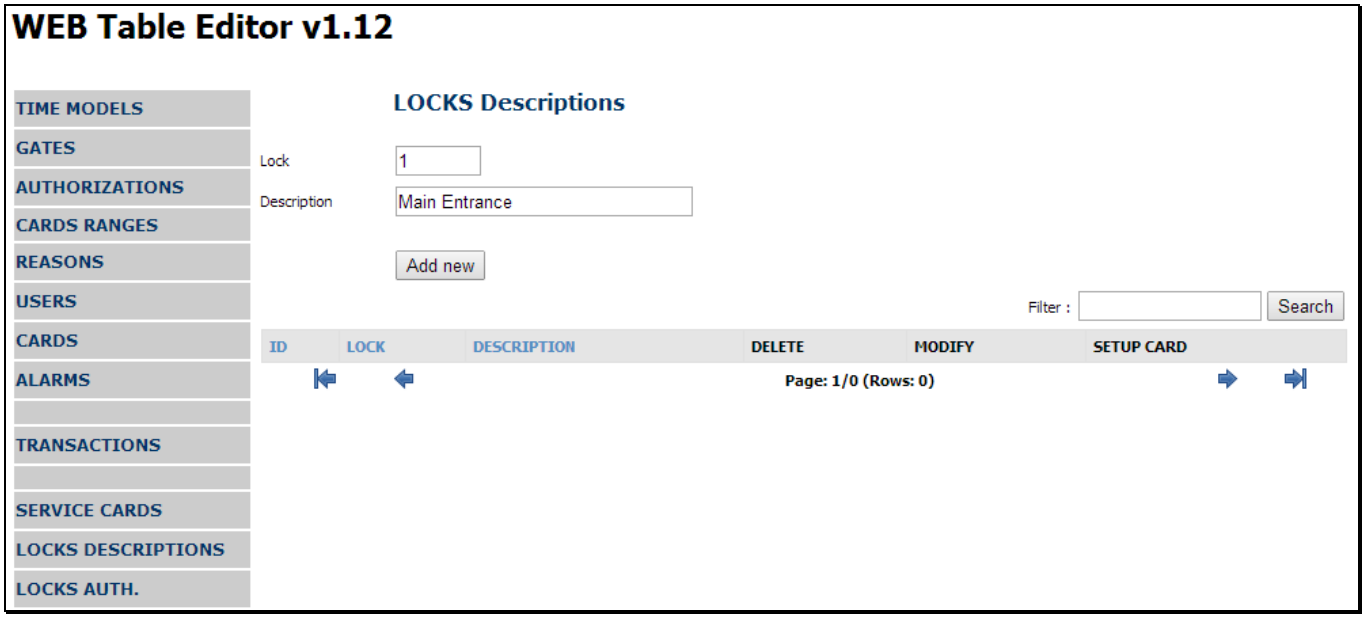
(*) The "standard sector" is the one which contains the information about the site code and the card validity start & end times; the "alarms sectors" are those which contain the information relevant to the events (typically, either the valid or not valid access attempts on any wireless locks and, in rare cases, internal lock events such as "void list full", "low battery", and so on); the "access matrix" sectors are those which contain the information about on which locks each card is enabled to access.

18.2 WIRELESS LOCKS INITIALIZATION

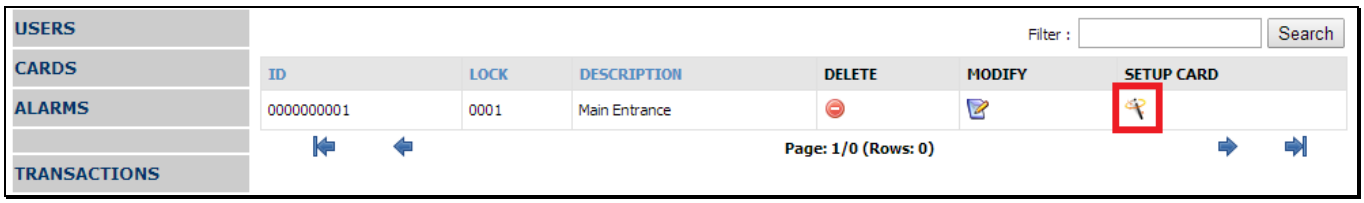
When you first receive a wireless lock out of the factory, it's not configured at all. The first thing to do is to open the wireless lock and insert the battery following the mounting instructions provided. After 2 seconds, the wireless lock LEDs should turn red for 1 second, then switch off for 6 seconds, then turn green for 1 second, then slowly blink yellow (red+green) for 20 seconds. At the end of this 30-seconds booting phase, the wireless lock is ready to be initialized in order to work in your specific offline Aperio wireless lock plant.

In particular, it should be assigned a "lock ID" (a unique identifier within the plant) and a "authorization group" number. Furthermore, it should be configured with the basic plant settings which are the same for all the wireless locks in the plant, i.e. the Mifare authentication keys and the specification of which sectors, inside the ACCESS CARDS to be used in the plant, are going to be used as "standard sector", "alarms sectors" and "access matrix".

In order to achieve that, you have to create a different LOCK (FIRST) SETUP CARD for each wireless lock, and that can be done by means of the "**LOCKS DESCRIPTIONS**" link at the left of each "**Web Table Editor**" page, which opens the following editor:



For each wireless lock used you should assign a different lock ID, starting from the first available ("1"), and manually enter it in the "Lock" text box (please note that, for simplicity reason, we always use a different authorization group for each wireless lock, thus each lock authorization group is always made up of a single wireless lock, and each authorization group number is always assumed equal to the single lock ID). You can then enter a custom lock description in the "Description" text box, in order to make it easier to remember where each wireless lock is located (and as a consequence, which ACCESS CARDS should be authorized on it). Then click the "**Add new**" button and click "OK" (or press <Enter>) in the confirmation prompt window to automatically store this information in a reserved file named **LOCKAUTHGRPDESC.TXT**: after this operation the web page will be automatically updated, showing a new entry in the locks list at the bottom:



At this point, if you click the "SETUP CARD" icon highlighted in red in the picture, the "**Lock Setup Card - Read your badge**" prompt will appear on the X1/X2 Aperio Plant Manager display, and the web page will wait for you to place an empty (unprotected) Mifare card on the reader. When you place the card (that must be done within 3 seconds), keep it still on the reader until the "**Operation completed**" confirmation message is shown on the X1/X2 Aperio Plant Manager display. In such case, the web page will show the "**Lock Setup card created**" confirmation message.

If, instead, something goes wrong, this may be due for several reasons:

1) you didn't place a Mifare card on the reader within 3 seconds. In such case no error messages are shown on the X1/X2 Aperio Plant Manager display after the "**Lock Setup Card - Read your badge**" prompt quits, while the web page will show the "**Unable to reading badge! See log file for more details.**" error message;
2) Either:
   - you didn't keep the Mifare card still for the time needed to complete the writing
   - you are using a Mifare card which is not empty (unprotected)

   In such cases the "**Operation failed**" error message is shown on the X1/X2 Aperio Plant Manager display at the end of the operation, while the web page will show the "**Unable to create Service Card!**" error message.

Assuming you have been able to successfully create the LOCK (FIRST) SETUP CARD you can then proceed with the wireless lock initialization, by placing this card directly on the wireless lock. Since this card also contains the current time stamp, in such a way to also set the wireless lock's internal clock (see details at §18.4 below), this should be done immediately after you have created it, in order to minimize the displacement between the time written on the card and the actual time at the moment you place it on the wireless lock. Anyway, shouldn't it be possible, you will be able to set the lock's internal clock afterwards (see §18.4 below).

When you place the LOCK (FIRST) SETUP CARD on the wireless lock, keep it still until the lock LEDs turn yellow (red+green) for 1 second, which happens after 2 seconds: from this moment on, the wireless lock is permanently configured with the unique site code and the corresponding Mifare authentication keys of your specific offline Aperio wireless lock plant, so can be used only with the protected Mifare cards created for this plant (if you try to place any other 13,56MHz smart card on it you wouldn't see any effect): also the LOCK (FIRST) SETUP CARD itself won't be read anymore by the same wireless lock once it has been configured, since actually it is an unprotected card whose purpose is to be used just once on a wireless lock with factory default settings.

For this reason, should you have several wireless locks to setup, you may also repeat the 3-steps procedure (1. Create a new lock entry by means of the "**LOCKS DESCRIPTIONS**" editor; 2. Write the corresponding LOCK (FIRST) SETUP CARD; 3. Place it on the wireless lock) for each wireless lock using the same Mifare card, which can be rewritten as many times as needed.

Warning: please note that even if the *[Aperio]* section of the PARAMETERS.TXT file (see §4.11 at page 60) was not properly configured (see §18.1 above for how to fix that) the LOCK (FIRST) SETUP CARD creation would succeed anyway, but in this case, of course, the corresponding lock would not be correctly configured.

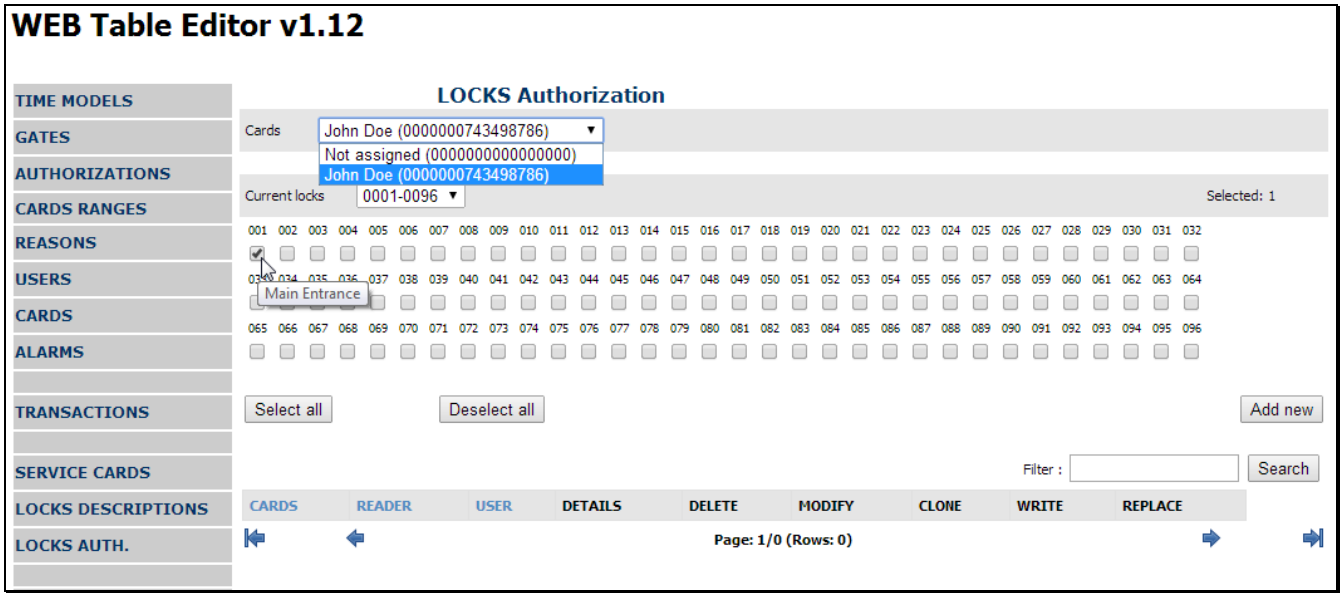
If you don't see any feedback when placing the LOCK (FIRST) SETUP CARD on the wireless lock, this means that the wireless lock is not in factory default conditions anymore: if you want to reset the lock to factory default, for example because you want to change its lock ID / authorization group, or start using it in a different offline Aperio wireless lock plant, you may create a FACTORY RESET CARD (see §18.4.2 at page 173). Note that, while the LOCK (FIRST) SETUP CARD is unprotected and can be used on any wireless lock with factory default settings, the FACTORY RESET CARD is protected and can be used only on the wireless locks belonging to its specific offline Aperio wireless lock plant. ***Note***: after you have successfully used a FACTORY RESET CARD on a wireless lock, please wait at least 20 seconds before placing a new LOCK (FIRST) SETUP CARD on it, otherwise it won't be recognized.

18.3 ACCESS CARDS MANAGEMENT

Once you have performed the wireless locks initialization, as described in the previous §18.2, you are ready to create (but only if your X1/X2 has been configured as an "Aperio Plant Encoder", by introducing the proper "APERIO ENCODER" FW activation key, see §4.12 at page 65) and manage the ACCESS CARDS to be assigned to the users, starting from empty (unprotected) Mifare cards. First of all, before managing its access authorizations on the wireless locks, each empty Mifare Card to be used as ACCESS CARD must have already been added to the CARDS.TXT file which represents the card codes white list. Optionally, by loading also the USERS.TXT file, you may also associate a user name to each card code, making it easier to manage the access authorizations. These preliminary operations can be easily done by means of the "**USERS**" and "**CARDS**" links at the left of each "**Web Table Editor**" page, as described at §5.14 at page 84.

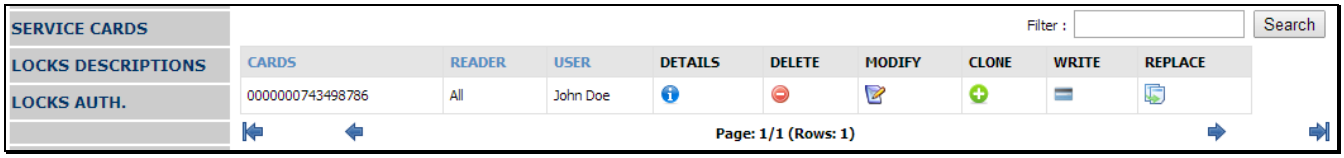
Notes: 1) while you are inside the "CARDS" editor, the 10-digit UID code of each ACCESS CARD can be entered directly from the embedded 13,56MHz Mifare R&W module, by selecting "**1 Primary**" from the drop-down menu at the right of the "Get from Reader" button, then pressing the button and reading the empty Mifare card within 5 seconds; 2) most of the users and cards attributes concerning the local access control logic (which could be applied for card readings performed on any reader physically connected to X1/X2 Aperio Plant Manager) are not considered at all for the ACCESS CARDS to be used on the offline Aperio wireless locks, so at this stage they can also be left at their default settings (the only relevant fields are the user names, the card codes and, optionally, the relevant start /end validity dates and times), and it doesn't matter whether the local access control logic is disabled or not.

If an empty Mifare card has already been added to the CARDS.TXT file, then you can proceed programming it as an ACCESS CARD and managing its access authorizations on the wireless locks, by selecting the "**LOCKS AUTH.**" link at the left of each "**Web Table Editor**" page, which opens the following editor:



In this case only one ACCESS CARD has been defined for the plant (the CARDS.TXT file only contains the card code "0743498786"), and this card has been optionally assigned to the user "John Doe", defined by means of the USERS.TXT file. So, by clicking on the "**Cards**" drop-down menu at the top you can find and select only this card code, with its user description (should USERS.TXT be missing you would see "undefined" instead of the user name).

Once selected a ACCESS CARD, then you can singly check / uncheck the numbers of one or more lock ID / authorization groups on which you want to respectively enable / disable this particular ACCESS CARD: as you can see in the picture above, in order to make it simpler to mentally associate the lock ID / authorization group to the physical wireless lock, the custom lock description which you have previously defined in the "**LOCKS DESCRIPTIONS**" editor (see §18.2 above) can be shown by simply placing the cursor on the relevant checkbox. You may also select all the authorization groups or deselect them all by means of the special buttons. The page only fits 96 authorization group checkboxes (those which can be managed with a single "access matrix" sector), so if needed you should select higher authorization group intervals by means of the "Current locks" drop-down menu. Finally, click the "**Add new**" button and click "OK" (or press <Enter>) in the confirmation prompt window to store this information in a reserved file named **LOCKAUTHGRP.TXT**: after this operation the web page will be automatically updated, showing a new entry in the ACCESS CARDS list at the bottom:



Immediately after that, a new prompt window will appear, asking "Do you want to write Lock Authorization also on badge?": if you click "OK" (or press <Enter>), then the "**Access Card - Read your badge**" prompt will appear on the X1/X2 Aperio Plant Manager display, and the web page will wait for you to place the empty (unprotected) Mifare card with the selected card code (UID) on the reader. When you place the card (that must be done within 3 seconds), keep it still on the reader until the "**Operation completed**" confirmation message is shown on the X1/X2 Aperio Plant Manager display. In such case, the web page will show the "**Writing badge done!**" confirmation message.

If, instead, something goes wrong, this may be due for several reasons:

1) the *[Aperio]* section of the PARAMETERS.TXT file (see §4.11 at page 60) is not properly configured (see §18.1 above for how to fix that): in this case nothing happens on the X1/X2 Aperio Plant Manager when you start the writing operation (the "**Access Card - Read your badge**" prompt is not shown at all), while the web page immediately shows the "**Unable to Write card!**" error message;
2) you didn't place a Mifare card on the reader within 3 seconds. In such case no error messages are shown on the X1/X2 Aperio Plant Manager display after the "**Access Card - Read your badge**" prompt quits, while the web page will show the "**Unable to writing badge! See log file for more details.**" error message;
3) Either:
   - you didn't keep the Mifare card still for the time needed to complete the writing
   - you are using a Mifare card with a different card code (UID) than the one expected

- you are trying to create a new ACCESS CARD from an empty (unprotected) Mifare card while the special "X1/X2 APERIO ENCODER" FW activation key (see §4.12 at page 65) has not been introduced (the simple re-writing of an already existing ACCESS CARD formatted for the same plant would succeed anyway): this case can be easily recognized by the

  "[2] [AperioCreateAccessCard] Missing Firmware Key Aperio!"

  record which is logged in the LOG.TXT file
- you are using a Mifare card which has already been formatted for a different offline Aperio wireless locks plant than the one for which this X1/X2 Aperio Plant Manager has been configured
- you are using a Mifare card which has already been formatted for the same plant but as a different type of card than a ACCESS CARD
- the embedded 13,56MHz Mifare R&W module for some reasons has not been configured yet / has a wrong configuration (see §18.1 above for how to fix that)

In all such cases the "**Operation failed**" error message is shown on the X1/X2 Aperio Plant Manager display at the end of the operation, while the web page will show the "**Unable to Write card!**" error message.

Anyway, the ACCESS CARD can also be created or re-written afterwards, once the corresponding entry has already been created: consider that re-writing a previously created ACCESS CARD is faster than creating a new one, since all the protected sectors ("standard sector", "alarms sectors" and "access matrix" sectors) have already been formatted, and you just need to re-write the lock authorizations in the "access matrix" sectors. If you don't want to change the previously set lock authorizations for that ACCESS CARD (which can be checked by clicking the "**DETAILS**" icon in the relevant row), just click the "**WRITE**" icon in the relevant row to immediately start the writing operation. Otherwise you can click the "**MODIFY**" icon in the relevant row to change the previously set lock authorizations for that ACCESS CARD. In this case, once you have done, click the "Apply changes" button to save the lock authorization settings and then proceed to a new ACCESS CARD writing operation, or click the "Cancel" button to quit without making changes.

When you place a ACCESS CARD on the wireless lock, its LEDs should turn green for 1 second, which means that the ACCESS CARD is authorized and the door is unlocked for 6 seconds (during this period the wireless lock won't perform any other reading). If, instead, the LEDS turn red for 1 seconds, this means that the ACCESS CARD is not authorized on that wireless lock, or its validity period has expired, or has not yet begun (check also §18.4.1 below).

With reference to this, X1/X2 Aperio Plant Manager also writes the correct start and end validity dates on each ACCESS CARD, according to the content of the "start card validity" and "end card validity" fields in the relevant record of the CARDS.TXT file, and of the "start user validity" and "end user validity" fields in the relevant user's record of the optional USERS.TXT file (if present). If both files are present, then the greater among the "start card validity" and "start user validity" dates, and the lesser among the "end card validity" and "end user validity" dates are written into the ACCESS CARD(*).

If you don't see any feedback when placing the ACCESS CARD on the wireless lock, this means that either the wireless lock has not been initialized (see §18.2 above) or has been configured for a different offline Aperio wireless locks plant.

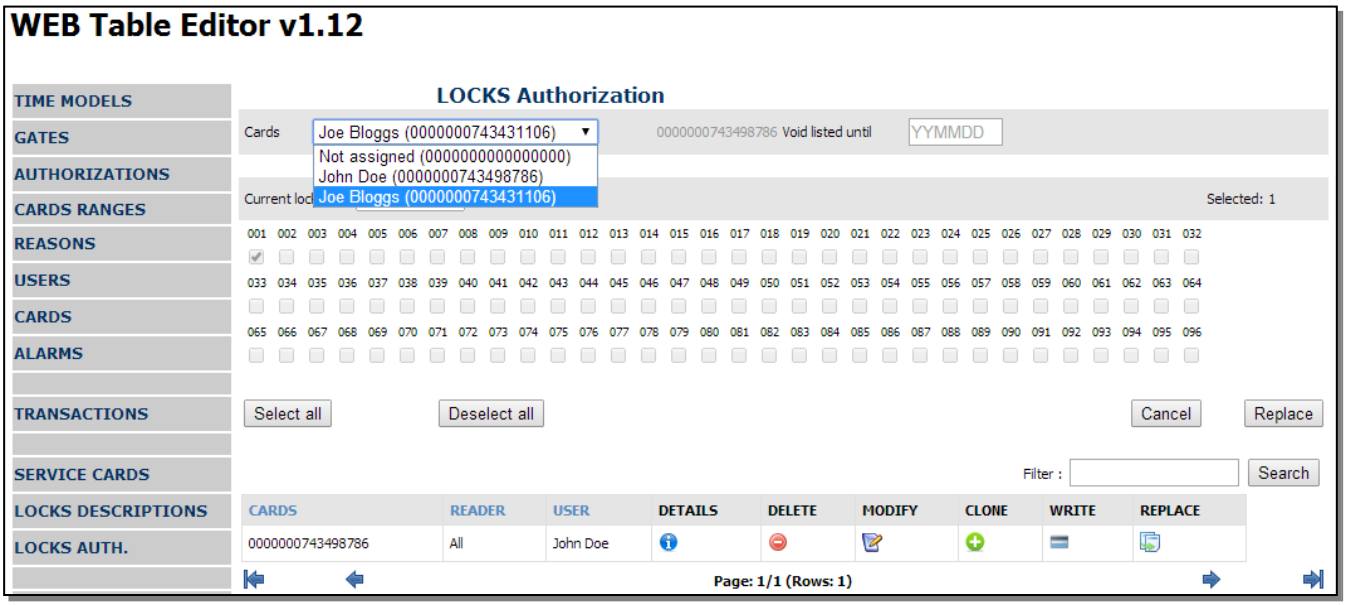
You can also "replace" an existing ACCESS CARD (which you cannot pick up from the relevant user) by creating a new one with the same lock authorizations, while at the same time putting the old card in a black list (here called "void list") which resides in each wireless lock.

Just click the "**REPLACE**" icon in the existing ACCESS CARD row, then proceed as follows:

1) select a new ACCESS CARD code by means of the "**Cards**" drop-down menu at the top (also in this case, of course, the new ACCESS CARDS to be created must have already been added to the CARDS.TXT file);

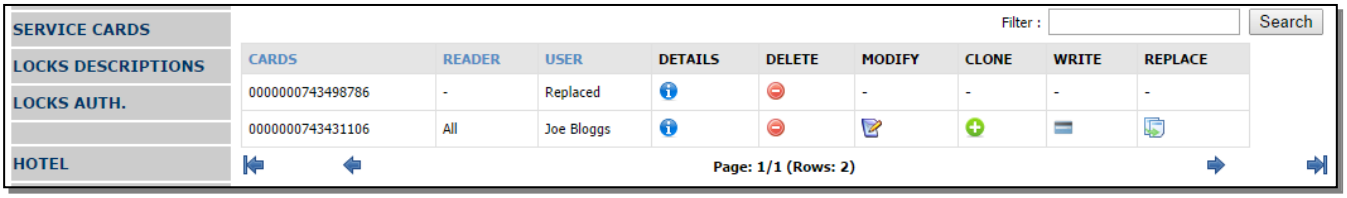
2) fill the "**Void listed until**" text box which refers to the old ACCESS CARD to be replaced, whose code is shown at its left (see picture blelow). The reason for this is that the void list residing on each wireless lock is limited to 100 records, and each of them cannot be cleared singularly, so depending on how many ACCESS CARDS are used in the plant and on how often they are replaced, the void list may be quickly filled: should that happen, the only available option is to completely clear the void list (see §18.4.2 below for how to do that). In order to avoid that, you should set the "**Void listed until**" field to a "not too far" future date, in such a way that the corresponding record in the void list will be automatically invalidated after that date, thus leaving a free space. Of course, if the old card is still in circulation, this may cause a security issue: for this reason, we recommend not to create ACCESS CARDS with undefined end validity dates (i.e. that will never expire, see the (*) note above). In facts, if you define a "not too far"

end validity date, it will be possible to set that same date in the "**Void listed until**" field, should that ACCESS CARDS be replaced afterwards, without creating a security issue;



3) As you can see in the picture, when replacing an ACCESS CARD the lock authorizations cannot be changed (the corresponding checkboxes are grayed out), so you can just click the "**Replace**" button to proceed, then click "OK" (or press <Enter>) in the confirmation prompt window;

4) The "**Access Card - Read your badge**" prompt will appear on the X1/X2 Aperio Plant Manager display, and the web page will wait for you to place the empty (unprotected) Mifare card with the new selected card code (UID) on the reader, according to the same procedure described above for the first creation of an ACCESS CARD: in case of success, the web page will show the "**Replaced card done!**" confirmation message, then it will update the ACCESS CARDS list at the bottom, by adding the new valid entry and invalidating the previous one (which will show "Replaced" instead of the user name):



5) The new ACCESS CARD created also contains, in its "standard sector", the specification of the old card code to be put in the void list, and of the relevant "**Void listed until**" field. In this way, whenever you'll use the new ACCESS CARD on a wireless lock of your offline Aperio wireless locks plant, this information will be transferred to that particular wireless lock, which will then create a new record in its resident void list. Of course, in order to make sure that the old ACCESS CARD could not be used anymore on any wireless lock of your plant, you should place the new ACCESS CARD on every single wireless lock on which the old card (and also the new, which always have the same lock authorizations) is actually authorized.

18.4 THE SPECIAL "APERIO" MENU

On a X1/X2 Aperio Plant Manager, the main supervisor menu (described at 10.5 at page 113) contains one more section: the special "Aperio" menu. While inside the main menu (remember that after 30 seconds of inactivity X1/X2 automatically exits from all menus), use the arrow keys ▲▼ to select the third item from the top ("Aperio") and ↵ to confirm.

The "Aperio" menu features several options that will be described in detail in the following. As always, use the arrow keys ▲▼ to select an option and ↵ to confirm.

18.4.1 WIRELESS LOCKS INTERNAL CLOCK SETUP

As we have seen at §18.2 above, during the wireless lock initialization, to be done by means of the LOCK (FIRST) SETUP CARD, the wireless lock's internal clock is also set.

Anyway, if you didn't place the LOCK (FIRST) SETUP CARD on the wireless lock immediately after you have written it, the lock's internal clock would be running late of the time passed between the LOCK (FIRST) SETUP CARD creation and the wireless lock initialization, since the time written on the card is different than the actual time at the moment you placed it on the wireless lock.

You may also realize that the time stamp written in the LOCK (FIRST) SETUP CARD, which always consists in the universal UTC/GMT time & date (in *Unix epoch time* format) calculated basing on the current local time & date as taken from the terminal clock (which must have been correctly set as described at §4.1 at page 16) and on the local time zone and the local DST settings (which must have been correctly configured in the *[TimeSettings]* section of the PARAMETERS.TXT file (see §4.11 at page 54, **UTC** parameter in particular), was actually wrong due to some misconfiguration.

Whatever the reason, you can set the lock's internal clock at any moment, even after the wireless lock initialization. In order to do that, anyway, you cannot use a LOCK (FIRST) SETUP CARD, since actually that is an unprotected card whose purpose is to be used just once on a wireless lock with factory default settings: you must create a different type of "lock setup card", which we'll call LOCK (TIME) SETUP CARD, which is protected (thus can be used only on any wireless lock which has already been configured for your specific offline Aperio wireless lock plant). Furthermore, the LOCK (TIME) SETUP CARD only contains the time stamp on the protected "standard sector", thus is much faster to be written compared to the LOCK (FIRST) SETUP CARD.

The LOCK (TIME) SETUP CARD can be created by selecting the second option ("**Sync Time & Date**") in the "Aperio" menu.

X1/X2 Aperio Plant Manager then opens a panel which lets you check the current local time & date and the universal UTC/GMT time & date (the only one which will be written in the LOCK (TIME) SETUP CARD), as calculated assuming the current terminal configuration is correct: should you see some inconsistency, please check the configuration in order to fix the problem before proceeding..

The "**Sync Time & Date**" panel also shows the permanent prompt "**Read your badge**", which has no timeout (you must press any button to exit). Thus, you can place an empty (unprotected) Mifare card on the reader whenever you want to create a new LOCK (TIME) SETUP CARD, and also re-write a previously created one as many times as needed to synchronize all the wireless locks in the same plant. When you place the card, keep it still on the reader until the "**Operation completed**" confirmation message is shown: consider that re-writing a previously created LOCK (TIME) SETUP CARD is faster than creating a new one, since the protected "standard sector" has already been formatted, and you just need to re-write the current time stamp in it.

If, instead, it shows the "**Operation failed**" error message, check §18.4.5 at page 175 in order to find the reason.

Assuming you have been able to successfully write the LOCK (TIME) SETUP CARD you can then proceed with the wireless lock internal clock setup, by placing this card directly on the wireless lock. This should be done immediately after you have written the LOCK (TIME) SETUP CARD by means of the "**Sync Time & Date**" panel, in order to minimize the displacement between the time written on the card and the actual time at the moment you place it on the wireless lock. For this reason, we recommend to perform this 2-steps procedure (1. Write the LOCK (TIME) SETUP CARD; 2. Place it on the wireless lock) by keeping both the wireless lock and the X1/X2 Aperio Plant Manager close at hand. This can be easily done as long as the wireless locks is

not yet mounted on the door, otherwise it will be necessary to move and power the X1/X2 Aperio Plant Manager in proximity of the door itself. Should you have several wireless locks to setup, we recommend to repeat the 2-steps procedure for each wireless lock: the "**Sync Time & Date**" panel allows to rewrite the same LOCK (TIME) SETUP CARD as many times as needed.

Important Note: the wireless lock internal clock setup procedure described above should also be repeated each time you replace a battery in a wireless lock, but only if the battery is missing for more than 30 seconds: in such case, in facts, the internal clock setting would be lost again, turning back to January 1st , 2007. A battery should allow 40000 card readings and last about 2 years. As the battery reaches the "low charge" threshold, the wireless lock LEDs start blinking yellow (red+green) each 5 seconds, while as it reaches the "very low charge" threshold they start blinking red each 5 seconds.

When you place the LOCK (TIME) SETUP CARD on the wireless lock, its LEDs should turn yellow (red+green) for 1 second, which means that the internal clock has been set.

18.4.2 SERVICE CARDS

- **AUDIT TRAIL CARD**

  If necessary, you can copy the last 200 lock events (including valid / not valid transactions, full void list and battery low events) on the card created with the third option ("**Audit Trail Card**") in the "Aperio" menu, by simply placing it on the wireless lock.

- **CLEAR VOID LIST CARD**

  The fifth option ("**Clear Void List**") in the "Aperio" menu (visible only by moving selection downwards until the next page) allows to create a card which may completely clear the black list resident on any lock of your plant, by simply placing it on the wireless lock.

- **RADIO ACTIVATION CARD**

  The sixth option ("**Radio Activ. Card**") in the "Aperio" menu (visible only by moving selection downwards until the next page) allows to create one more type of protected service card called RADIO ACTIVATION CARD, which can be used on any wireless lock which has already been configured for your specific offline Aperio wireless lock plant: the purpose of this card is to temporarily activate the wireless lock radio communication, in such a way that it can be detected and configured by a special *Aperio Service Tool*, which consists of a USB radio dongle and a *Offline Programming Application (PAP)*. Please note that the Aperio Service Tool is not included in the offline wireless lock kit, and must be purchased separately when needed, for example if you want to create a plant with more than 8 total "alarms" + "access matrix" sectors on each ACCESS CARD (see note at §18.1 at page 166), or if you want to check (or restore to factory default) the current configuration of a wireless lock that's already been initialized for an unknown offline Aperio wireless lock plant. Please refer to the relevant documentation supplied by Assa Abloy for further information.

  Note: first install the offline PAP tool, then insert the USB dongle, and when you will be asked for how to find a suitable device driver you should manually select the "Tritech TriBee USB Driver" subfolder under the program installation folder (usually "C:\Program Files (x86)\Assa Abloy\Aperio Offline Programming Application").

  If you don't have the offline PAP tool, then the RADIO ACTIVATION CARD is not needed at all and should not be used, anyway if you place it on the wireless lock its LEDs should turn yellow (red+green) for 2 seconds, then start blinking yellow each second for 1 minute (within the whole period the wireless lock could be detected by the PAP Tool).

- **FACTORY RESET CARD**

  The seventh option ("**Clear Void List**") in the "Aperio" menu (visible only by moving selection downwards until the next page) allows to create a card which may restore the factory default settings (no Mifare authentication keys & sectors definition, no LOCK ID / authorization group) on any already initialized lock of your plant, by simply placing it on the wireless lock.

Note: for any option described in this paragraph, the relevant panel shows the permanent "**Read your badge**" prompt, which has no timeout (you must press any button to exit). Thus, you can place an empty (unprotected) Mifare card on the reader whenever you want to create a new service card, and also re-write a previously created one of the same type. When you place the card, keep it still on the reader until the "**Operation completed**" confirmation message is shown (this may take several seconds).

If, instead, it shows the "**Operation failed**" error message, check §18.4.5 at page 175 in order to find the reason.

18.4.3 CHECKING AND ERASING CARDS

The first option ("**Card Info**") in the "Aperio" menu allows to check the type and validity period of any card formatted for the same offline Aperio wireless locks plant for which this X1/X2 Aperio Plant Manager has been configured.

The "**Radio Activ. Card**" panel shows the permanent "**Read your badge**" prompt, which has no timeout (you must press any button to exit). Thus, you can place a card on the reader whenever you want, keeping it still until the card check result is shown, which can be one among the following (this information is temporarily shown for 10 seconds -during which you may also place on the reader another card to be checked- then the "**Radio Activ. Card**" panel automatically quits):

- **Access card – Start** *YYYY-MM-DD HH:MM:SS* **– End** *yyyy-mm-dd hh:mm:ss*

  Note: if the start or end validity dates have not been specified (that is equivalent to a field left at "0000000000" in the CARDS.TXT and USERS.TXT files, which means "always"), only the "-" character is shown in place of them.

- **Lock setup card – Start - – End -**

  Note: this message appears for both types of "lock setup card": the unprotected LOCK (FIRST) SETUP CARD and the protected LOCK (TIME) SETUP CARD; the "Start" and "End" fields are always empty since even if these cards could be assigned a validity period, X1/X2 Aperio Plant Manager creates them as "always valid".

- **Audit trail card** *or* **Clear void list Card** *or* **Radio activ. Card** *or* **Reset card**

  Note: all these types of card don't have a validity period, so it is not shown.

- **Empty Card**

  Note: this message appears for any unprotected card which is not provided with a offline Aperio wireless locks plant segment configuration.

If, instead, it shows the "**Operation failed**" error message, check the following §18.4.5 in order to find the reason.

The ninth and last option ("**Erase Card**") in the "**Aperio**" menu (visible only by moving selection downwards until the third page) allows to completely remove the offline Aperio wireless locks plant configuration from any type of card which is currently formatted for the same plant for which this X1/X2 Aperio Plant Manager has been configured.

The "**Erase Card**" panel shows the permanent "**Read your badge**" prompt, which has no timeout (you must press any button to exit). Thus, you can place a card on the reader whenever you want, keeping it still until the "**Operation completed**" confirmation message is shown (this may take several seconds). In such case, it will be possible to use the card erased as any unprotected general purpose Mifare card.

If, instead, it shows the "**Operation failed**" error message, check the following §18.4.5 in order to find the reason.

18.4.4 OTHER OPTIONS

The fourth option ("**Access Card**") allows preliminary formatting of an emtpy (unprotected) Mifare card as an ACCESS CARD for your plant, but without defining the locks authorizations (it wouldn't open any lock).

The "**Access Card**" panel shows the permanent "**Read your badge**" prompt, which has no timeout (you must press any button to exit). Thus, you can place an empty (unprotected) Mifare card on the reader whenever you want to create a new

access card, and also re-write a previously created one of the same type. When you place the card, keep it still on the reader until the "**Operation completed**" confirmation message is shown (this may take several seconds).

If, instead, it shows the "**Operation failed**" error message, check §18.4.5 at page 175 in order to find the reason.

The eight option ("**Site Code**") allows to change the site code: this makes it possible to use the same X1/X2 to manage different offline Aperio wireless locks plants (of course one at a time: as soon as you change the site code, it won't be possible to manage the previous plant anymore, unless you restore the previous site code).

The "**Site Code**" panel shows the current 10-digits site code in decimal format: use the ▲▼ keys to modify a single digit (on X2 only, you can also press the corresponding key on the numeric keyboard), ↵ (Enter) to move selection to the next digit or to confirm while you're on the last one and **Clr** to go back to the previous position or to abort while you're on the first (there's a 30 seconds inactivity timeout). We recommend to change only the least significant digits, in order to use progressive numbers for the various plants.

18.4.5 TROUBLESHOOTING

Should you incurr in the "**Operation failed**" error message when trying to read (in case of the "**Card Info**" option only) or write any type of protected service cards within the "Aperio" menu, it may be due to one among the following reasons:

1) you didn't keep the Mifare card still for the time needed to complete the reading / writing;
2) you are using a Mifare card which has already been formatted for a different offline Aperio wireless locks plant than the one for which this X1/X2 Aperio Plant Manager has been configured;
3) you are using a Mifare card which has already been formatted for the same plant but as a different type of service card than the one you're trying to read / write;
4) the *[Aperio]* section of the PARAMETERS.TXT file (see §4.11 at page 60) is not properly configured, or the embedded 13,56MHz Mifare R&W module for some reasons has not been configured yet / has a wrong configuration (see §18.1 above for how to fix that).

19. STRUMENTI SOFTWARE

Uno dei principali vantaggi di usare protocolli standard e file di testo è che potete usare dei software standard comunemente disponibili per i test e la programmazione.

Non sono necessari DLL o SDK proprietari, o strumenti specifici.

Per configurare il terminale da remoto potete collegarvi al suo server web integrato, usando un comune browser come:

Firefox (http://www.mozilla-europe.org/it/),
Internet Explorer (http://windows.microsoft.com/it-IT/internet-explorer/downloads/ie),
Google Chrome (http://www.google.com/chrome/)…

già installato su qualunque PC.

Dovete solo digitare l'IP del terminale nella barra degli indirizzi del vostro browser preferito e navigare nel sito web del terminale.

Per configurare il terminale è anche possibile usare un programma client HTTP che invia opportuni comandi in risposta ai messaggi "*Keep Alive*" ricevuti dal terminale. Si veda il §12.3 a pag. 140 per approfondire il concetto di messaggio "*Keep Alive*", ed il §12.4 a pag. 141 per sapere quali devono essere il formato della risposta del client HTTP ed i comandi di configurazione disponibili.

Per sviluppare un programma che riceve i messaggi "*Keep Alive*" e le transazioni in online da X1/X2 vi basta usare delle librerie HTTP standard come la HTTPListener class del .NET framework.

Dalla sezione "Utility & SW" dell'area partners di Zucchetti AXESS (http://www.axesstmc.com/it/partnersarea/) potete scaricare "X1HTTPDemo": un semplice programma demo in .NET con codice sorgente che implementa un esempio di server web pronto a ricevere a rispondere ai messaggi online HTTP GET generati da X1/X2 in modalità online (**Attenzione**: riconosce solo le transazioni in formato standard, vedi §7 a pag. 96).

La configurazione può anche essere effettuata caricando file di testo via FTP, direttamente sul file system del terminale nella sua micro-SD card.

Il terminale è un server FTP , quindi vi bastano uno dei tanti programmi client FTP disponibili per il download (con l'esclusione di FireFox FireFTP), come ad esempio FileZilla:

http://filezilla-project.org/download.php?type=client

oppure anche le funzionalità client FTP integrate del vostro sistema operativo, navigando su ftp://< *Indirizzo_IP_Terminale*> oppure usando il comando "ftp <*Indirizzo_IP_Terminale*> " dalla finestra del prompt dei comandi (sono a disposizione, su richiesta, dei semplici file *batch* per effettuare lo scarico dei dati via FTP in maniera automatica da uno o più terminali X1/X2). col nuovo firmware, uscendo quindi dal menu USB.

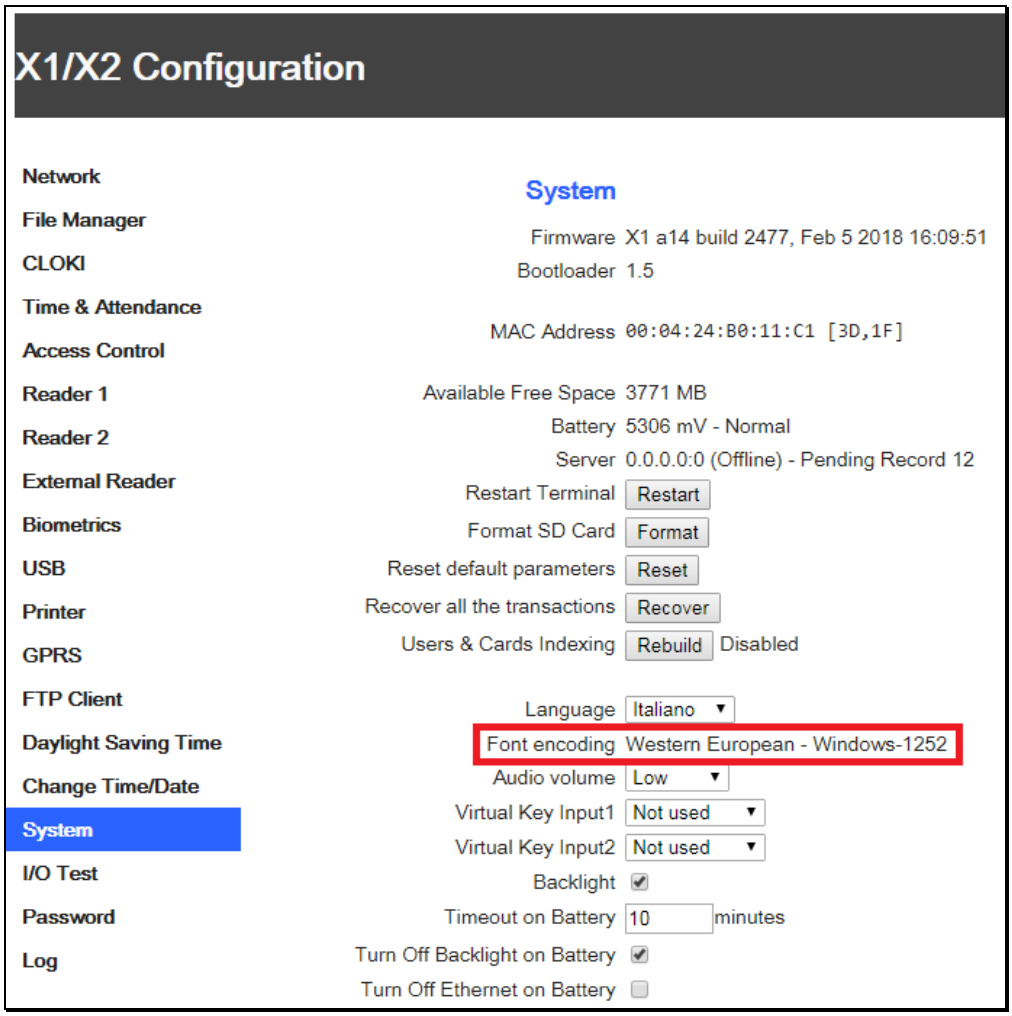
Per utilizzare le funzionalità di client FTP del terminale è necessario creare un server FTP ed una utenza autorizzata all'invio dei dati. È possibile creare il server FTP utilizzando le funzionalità integrate di alcuni sistemi operativi oppure uno dei programmi server FTP disponibili per il download, come ad esempio FileZilla Server:

http://filezilla-project.org/download.php?type=server

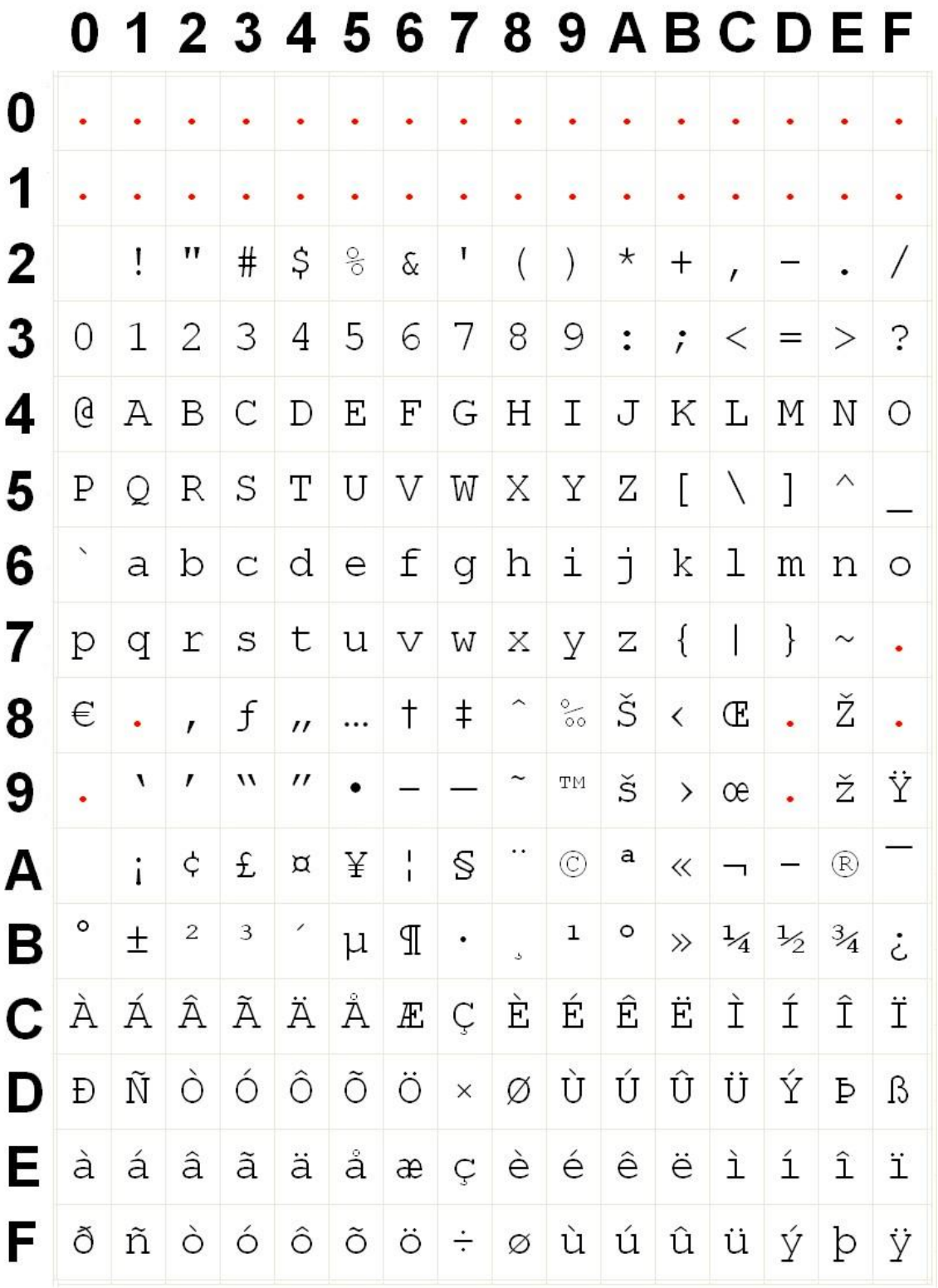
20. MAPPE DEI CARATTERI

A partire dalla versione a10_build715, il firmware standard di X1/X2 può visualizzare solo i caratteri inclusi nella tabella di codifica Windows-1252, relativa al set di caratteri Europa occidentale: non è più possibile selezionare un diverso set di caratteri mediante il parametro **FontEncoding** all'interno della sezione *[System]* del file PARAMETERS.TXT (parametro che viene comunque mantenuto per retrocompatibilità), e non è più disponibile il relativo menu a tendina nella pagina "**System**" del web server HTTP del terminale.

Se avete necessità di utilizzare un diverso set di caratteri dovete caricare una delle versioni speciali di firmware *pppp_VNN_***build***nnn*, contenenti ciascuna solo i caratteri inclusi nella tabella di codifica Windows-*pppp*: sono disponibili le versioni 1254 (Turco), 1250 (Europa centro-orientale), 1251 (Russo / Cirillico), 1253 (Greco). Nella pagina "**System**" del web server HTTP del terminale potete verificare alla voce **Font encoding** quale sia il set di caratteri attualmente disponibile:

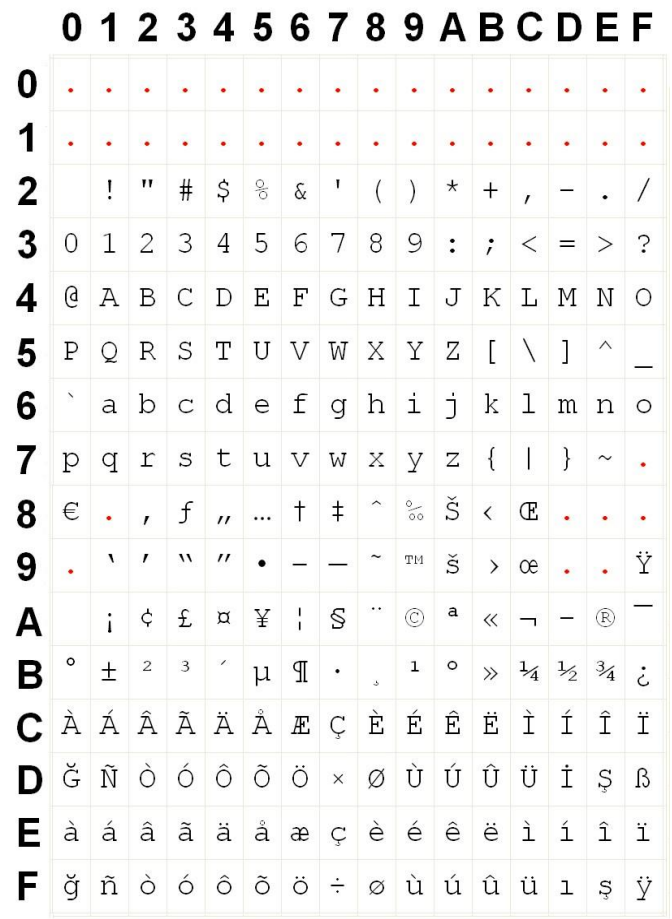


Nella pagina seguente sono mostrate le tabelle di codifica Windows-125*x* relative a ciascun set di caratteri (in cui sono riportati gli indici dei caratteri in valore esadecimale); i punti rossi si riferiscono a caratteri inutilizzati o di controllo.



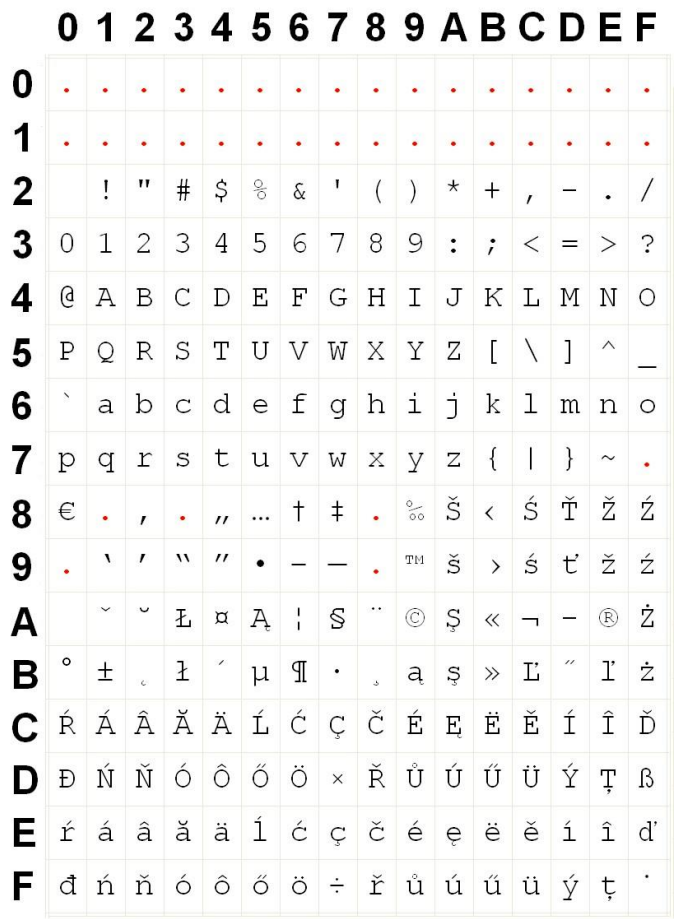
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2		!	"	#	$	%	&	'	(	)	*	+	,	–	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	.
8	€	.	‚	ƒ	„	…	†	‡	ˆ	‰	Š	‹	Œ	.	Ž	.
9	.	‘	’	“	”	•	–	—	˜	™	š	›	œ	.	ž	Ÿ
A		¡	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	–	®	¯
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

Europa occidentale (Windows-1252)
Firmware standard *VNN*_**build***nnn*



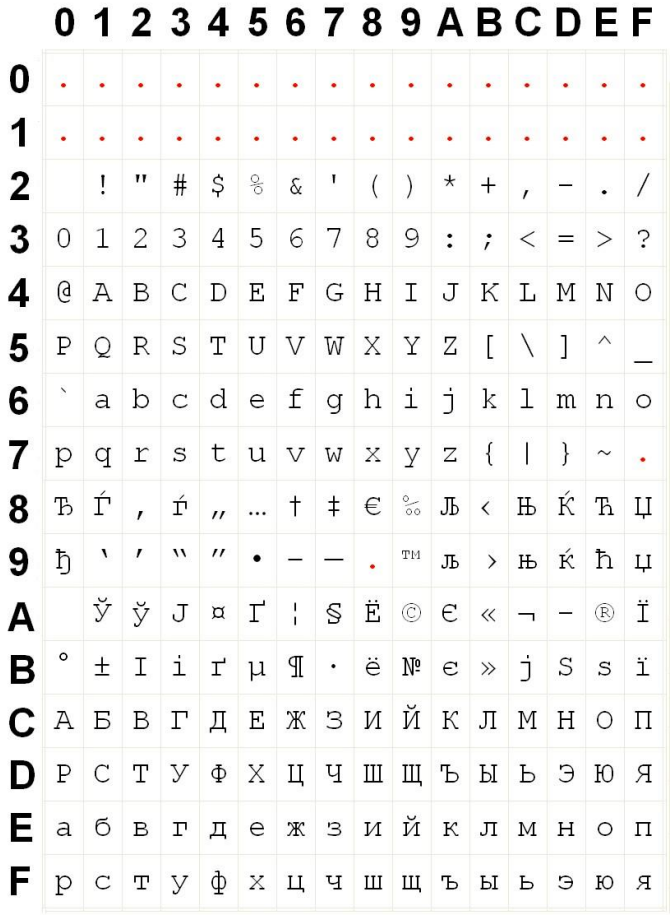
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	.
8	€	.	‚	ƒ	„	…	†	‡	ˆ	‰	Š	‹	Œ	.	.	.
9	.	‘	’	“	”	•	–	—	˜	™	š	›	œ	.	.	Ÿ
A		¡	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	-	®	¯
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ğ	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	İ	Ş	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ğ	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ı	ş	ÿ

Turco (Windows-1254)
Firmware **1254_***VNN*_**build***nnn*



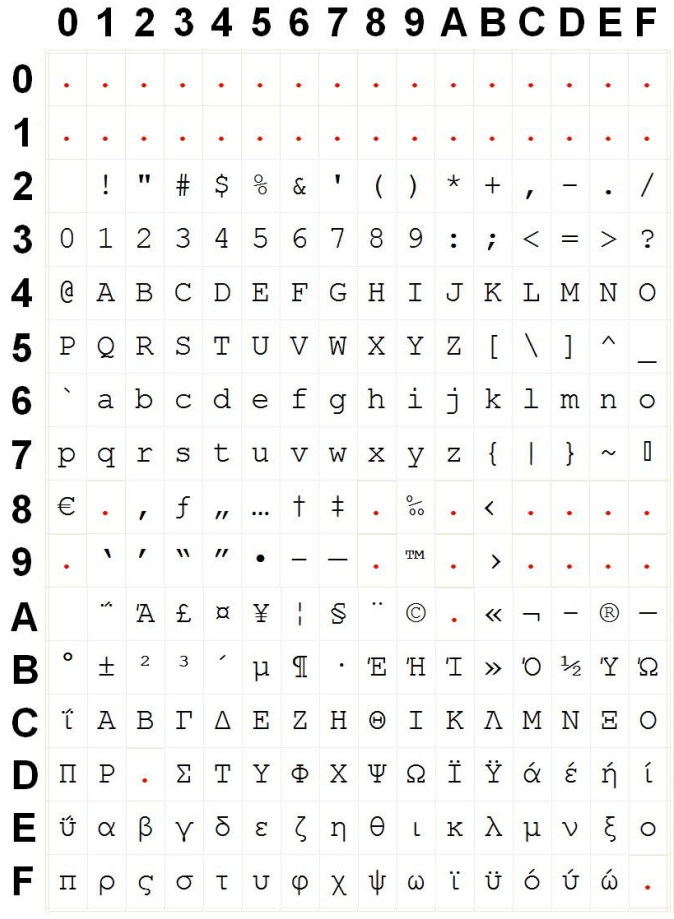
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	.
8	€	.	‚	.	„	…	†	‡	.	‰	Š	‹	Ś	Ť	Ž	Ź
9	.	‘	’	“	”	•	–	—	.	™	š	›	ś	ť	ž	ź
A		ˇ	˘	Ł	¤	Ą	¦	§	¨	©	Ş	«	¬	-	®	Ż
B	°	±	˛	ł	´	µ	¶	·	¸	ą	ş	»	Ľ	˝	ľ	ż
C	Ŕ	Á	Â	Ă	Ä	Ĺ	Ć	Ç	Č	É	Ę	Ë	Ě	Í	Î	Ď
D	Đ	Ń	Ň	Ó	Ô	Ő	Ö	×	Ř	Ů	Ú	Ű	Ü	Ý	Ţ	ß
E	ŕ	á	â	ă	ä	ĺ	ć	ç	č	é	ę	ë	ě	í	î	ď
F	đ	ń	ň	ó	ô	ő	ö	÷	ř	ů	ú	ű	ü	ý	ţ	˙

Europa centro-orientale (Windows-1250)
Firmware **1250_***VNN*_**build***nnn*



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	.
8	Ъ	Ѓ	‚	ѓ	„	…	†	‡	€	‰	Љ	‹	Њ	Ќ	Ћ	Џ
9	ђ	‘	’	“	”	•	–	—	.	™	љ	›	њ	ќ	ћ	џ
A		Ў	ў	Ј	¤	Ґ	¦	§	Ё	©	Є	«	¬	-	®	Ї
B	°	±	І	і	ґ	µ	¶	·	ё	№	є	»	j	S	s	ї
C	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
D	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
E	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
F	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я

Russo / Cirillico (Windows-1251)
Firmware **1251_***VNN*_**build***nnn*



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	▯
8	€	.	‚	ƒ	„	…	†	‡	.	‰	.	‹	.	.	.	.
9	.	‘	’	“	”	•	–	—	.	™	.	›	.	.	.	.
A		΅	Ά	£	¤	¥	¦	§	¨	©	.	«	¬	-	®	―
B	°	±	²	³	΄	µ	¶	·	Έ	Ή	Ί	»	Ό	½	Ύ	Ώ
C	ΐ	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ξ	Ο
D	Π	Ρ	.	Σ	Τ	Υ	Φ	Χ	Ψ	Ω	Ϊ	Ϋ	ά	έ	ή	ί
E	ΰ	α	β	γ	δ	ε	ζ	η	θ	ι	κ	λ	μ	ν	ξ	ο
F	π	ρ	ς	σ	τ	υ	φ	χ	ψ	ω	ϊ	ϋ	ό	ύ	ώ	.

Greco (Windows-1253)
Firmware **1253_***VNN*_**build***nnn*

Correzioni del documento dalla versione X1-X2 Manuale Utente r.1.1:

- Aggiunte opzioni lettori HID al §2
- Aggiunta nota sulla cancellazione di PARAMETERS.TXT al §4.7 e al §4.8
- Aggiornate descrizioni parametri RelayActivation e CardDecode al §4.8
- Aggiunte descrizioni nuovi parametri CustomRecord, CustomEntry, CustomExit, TimeLock, FtpPort al §4.8
- Aggiornato elenco parametri di default al §4.8
- Aggiunta nota sul formato personalizzato di TRANSACTIONS.TXT al §6
- Aggiunto §6.1 per la definizione del formato personalizzato di TRANSACTIONS.TXT
- Aggiunte note sui file batch per lo scarico automatico via FTP al §6 e al §11
- Aggiornato file LANGUAGE.TXT di default al §7
- Aggiornate schermate ai §4 e §7

Correzioni del documento dalla versione X1-X2 Manuale Utente r.1.2:

- Cambiato nome segnale +VDCIN e immagine scheda al §3.2
- Cambiato nome campo “rele1” in “relay1” al §10.2
- Aggiunto campo “show” al §10.2
- Rimosso parametro RecordSource al §4.8 e aggiunto campi sempre presenti RFU1, RFU2 e SOURCE al §6
- Aggiornato elenco parametri e relativi valori di default al §4.8
- Aggiunte codifiche 125KHz Dating “4° nibble” e Kronotech + varie opzioni Wiegand al parametro CardDecode, §4.8
- Aggiunta pinatura Wiegand al §3.2
- Aggiornate schermate al §4, §4.1 e §7
- Cambiata descrizione del file DATETIME e aggiunte note al §4.1
- Aggiunti §3.5 e §3.7 sul collegamento dei lettori e il controllo della versione di fw
- Aggiornate diverse caratteristiche tecniche al §2
- Aggiunte inversione direzione e digitazione manuale ai pulsanti attivi in stand-by al §9.2
- Aggiunta digitazione manuale al §9.3
- Aggiunti §11 e §12 per il funzionamento a batteria e il trasferimento dati via USB
- Il vecchio §11 è diventato §13
- Aggiunti campi $batt$ e $battmV$ al §10.1
- Introdotto nuovo §9.4 e scalati tutti i paragrafi seguenti
- Aggiunto sottomenu “Server” al §9.5
- Cambiata descrizione campo SOURCE al §6
- Aggiunto identificatore di campo “S” al §6.1
- Modificate alcune frasi sparse per migliorare la comprensibilità
- Cambiata durata ricarica e funzionamento a batteria al §3.2
- Aggiunta nota su aggiornamento firmware via USB al §8
- Modificata descrizione del par. CompanyName al §4.8
- Aggiunta nota sul file UPDATECONF.TXT al §4.7
- Aggiunte note e immagine File Manager su web server http al §4

- Unificati §4.5 e 4.6 in §4.6 e creato nuovo §4.7 per la gestione remota dei relé da web server http
- Aggiunta nota per invio comando seriale TTL da web server http al §4.6
- Aggiunto §3.6 per la scheda di espansione opzionale 914 NeoMAX
- Aggiunti campi cmd=… e file=… con esempi al §10.4
- Aggiunto campo pin=… al §10.2
- Aggiunta codifica HID iClass seriale al parametro CardDecode, §4.8
- Aggiunte note al §3.3 per i relé aggiuntivi su scheda 914 NeoMAX
- Completato §5 per il controllo degli accessi
- Aggiunta mappa dei caratteri al nuovo §14
- Aggiornato elenco messaggi in lingua al §7
- Aggiunte schermate con nome utente al §9.3
- Aggiunta schermata e note su AXREASON.TXT al §9.6
- Cambiate schermate e aggiunte note su AXREASON.TXT e REASONS.TXT al §9.7
- Rimossa nota "solo X2" in tutti i punti relativi alla gestione della porta USB
- Aggiunta nota per inserimento manuale di data e ora al §4.1
- Aggiunto sottomenu "Time & Date" al §9.5 e aggiornate schermate
- Aggiunto nuovo §4.5 per la selezione diretta della causale (file FKEY.TXT)
- Aggiunta modalità "solo PIN" e selezione diretta di una causale al §9.2
- Aggiunta nota sulla selezione diretta della causale al §9.6

Correzioni del documento dalla versione X1-X2 Manuale Utente r.1.3:

- I lettori barcode sono ora supportati al 2, figura al §3.2, §3.5, e nella descrizione del parametro CardDecode al §4.8
- Il PIN al §9.4 è ora mascherato (aggiunta nota e cambiata figura)
- Aggiunta descrizione nuovi campi CONTROLLI e ESITO al §6
- Aggiornato file LANGUAGE.TXT con nuove lingue spagnolo, francese e tedesco e aggiornata immagine al §7
- Aggiunte note per la disattivazione della revisione dati locale al §9.2 e §9.7
- Aggiunto tasto di selezione del menu ridotto relativo ai tasti di selezione diretta al §9.2
- Aggiunto §9.8 per il menu ridotto relativo ai tasti di selezione diretta
- Aggiunte note per il menu ridotto relativo ai tasti di selezione diretta e cambiato formato file e esempio al §4.5
- Aggiunta descrizione posizionamento su causale in base ai tasti premuti al §9.6
- Aggiunta descrizione uso dei parametri BeepOk, BeepError, ShowCardCodeBegin e ShowCardCodeLength al §9.3
- Cambiato valore di default del par. Contrast=2 al §4.8
- Aggiunta nota sulla porta UDP 8499 al §3.4
- Cambiata numerazione dei paragrafi (c'erano due §4.7)
- Aggiunto nuovo paragrafo §4.11 per l'attivazione di funzioni opzionali del firmware
- Aggiunta nota sull'invio di comandi in tempo reale forzando il keepalive al §10.3
- Rimosse indicazioni relative all'uso delle HTTP GET per configurare il terminale dai §4 (punto 4), §4.1, §4.8 e §13, e sostituite con quelle relative all'uso dei comandi in risposta ai messaggi "Keep Alive"
- Aggiunta nota sul file btransactions.loc al §9.7
- Aggiunti caratteri '&' alle stringe inviate in risposta ai comandi su KeepAlive al §10.4

- Aggiunti comandi RECOVER, GETPAR e SETPAR al §10.4
- Aggiunto nuovo §6 per la gestione del varco e cambiata numerazione capitoli successivi
- Indicato valore di default del par. Offline al §4.8
- Aggiunta nota su X1HTTPDemo al §14
- Aggiunta descrizione funzionamento file btransactions.loc al §7
- Gestione input da scheda NeoMAX e modulo biometrico FingerBOX ora disponibili al §2 e §3.6
- Aggiunti 2 input digitali di serie al §2
- Aggiunto nuovo §3.4 per gli ingressi digitali e cambiata numerazione paragrafi successivi
- Modificata figura scheda al §3.2 e aggiunta indicazione ingressi digitali
- Il flag 'T' nel file USERS al §5.9 è ora gestito
- Modificate descrizioni parametri RelayActivation e EntryRelay al §4.9
- Aggiunto §7.2 per i record di TRANSACTIONS.TXT relativi a emissione/rientro eventi
- Cambiato titolo e schermata al §4.7, aggiunta descrizione stato input digitali e ulteriore schermata
- Aggiunte specifiche FingerBox al §2
- Il FingerBOX è ora supportato al §3.6
- Aggiornate tutte le schermate relative al web server HTTP
- Aggiunta nota sulle cartelle BIOEXP e BIOIMP al §4
- Aggiunto inserimento manuale codici in revisione dati al §10.7
- Aggiunto comando TMC-UDP "V" e descrizione keep alive UDP al §3.5
- Aggiunta nota sul keep alive UDP al §11.3
- Aggiunte note sull'abilitazione del protocollo HTTP ai §4, 6.5 e 11
- Aggiunti descrizione diversi parametri al §4.9
- Aggiornato contenuto file PARAMETERS.TXT al §4.9
- Cambiata descrizione del parametro RelayActivation al §4.9
- Cambiata schermata e descrizione inserimento PIN al §10.4
- Cambiata schermata richiesta password e aggiunto schermata menu configurazione al §10.5
- Aggiunta schermata menu Ethernet e sottomenu MasterURL + note relative al §10.5
- Aggiornata descrizione parametro MasterURL al §4.9
- Aggiunta schermata menu Enable USB e descrizione al §10.5
- Aggiunta nota menu Enable USB al §13
- Cambiata frase al §5.2
- Aggiunta nota su par.AskPin al §5.9
- Aggiunti parametri AskPin e Firmware al §4.9
- Aggiunto nuovo §4.6 per DIRECTION.TXT e cambiata numerazione dei capitoli seguenti in 4.7, 4.8, 4.9, 4.10, 4.11
- Cambiata descrizione del parametro DirMode al §4.10
- I lettori a tripla traccia sono ora supportati al §2
- Aggiunti valori per la decodifica carte in tripla traccia al par. CardDecode al §4.10
- Aggiunto parametro FullTable al §4.10
- Aggiunta funzione "recover" delle timbrature al §7

- Aggiunto nuovo §11 per la gestione del FingerBOX e rinumerati capitoli seguenti
- Aggiunta nota al parametro TrnsFileUSB al §4.10
- Il limite per il par.CustomFormat è ora di 68 caratteri al §4.10 e 7.1
- Aggiunta autenticazione biometrica al §10.2
- Cambiato titolo e frase al §10.3
- Aggiunta nota sugli utenti amministratori al §10.5
- Aggiunta schermata con opzioni biometria al menu USB al §14
- Aggiunti §14.5 e 14.6 per salvataggio / importazione biometria con chiavetta USB
- Cambiata descrizione valore 30 del par. CardDecode e aggiunto valore 32 per codifica custom TMC Mifare
- Aggiunte note sul valore massimo del campo ora nei file di controllo accessi al §5
- Cambiata descrizione del campo SOURCE al §7

Correzioni del documento dalla versione X1-X2 Manuale Utente r.1.4:

- Modificata nota2 al §4.1: il file DATETIME viene ora automaticamente cancellato
- Aggiunta schedulazione dello scarico del file TRANSACTIONS.TXT via client FTP al §4.2
- Aggiunta sezione [FtpClient] al §4.10
- Aggiunte note sullo scarico del file TRANSACTIONS.TXT via client FTP e penna USB al §7
- Aggiunto paragrafo relativo al client FTP §7.3
- Aggiunte note su come creare un server FTP su PC al §15
- Modificate schermate col nuovo messaggio “Rimuovere dito” al §10.2
- Modificato messaggio di errore “data non valida” a “tessera scaduta” al §5.13 e descrizione codice di errore al §7
- Inserito nuovo capitolo §15 per le gestione del modem GPRS e cambiata numerazione dei cap. successivi
- Cambiata descrizione web server al §8
- Aggiunto parametro FontEncoding al §4.10
- Aggiunte mappe caratteri Turco e Europa centro-orientale e cambiata descrizione al §17
- Il modem GPRS opzionale è ora disponibile fra le opzioni al §2, in alternativa al lettore esterno
- Aggiunta nota sull’utilizzo del lettore esterno in alternativa al modem GPRS al §3.6
- Aggiunta nota sull’icona stato di carica batteria al §13
- Aggiunta descrizione parametri nella sezione [GPRS] al §4.10
- Aggiunta nota sul segnalatore acustico regolabile al §2
- Aggiunti parametri AudioVolume e TTY1Legacy al §4.10

Correzioni del documento dalla versione X1-X2 Manuale Utente r.1.5:

- Aggiunta nota su eventuali conflitti elettrici col lettore esterno al §15
- Estesa descrizione icona al §15.1
- Variata descrizione al §3.6
- Aggiunta nota sui caratteri alfanumerici nei codici utente in seguito a lettura di carta ai §3.6 e §10.2
- Aggiunta note sulle carte Mifare contenenti template da 256 byte al §10.2
- Aggiunte icone e logo personalizzato e relative note al §10.2
- Cambiata descrizione campi “del” e “file” al §12.4

- Aggiunto server tag “$fullcode$” al §12.1
- Aggiunta nota al parametro LogLevel al §4.10
- Aggiunte nuove codifiche Byte, BCD e lettore seriale TTL generico al parametro CardDecode al §4.10
- Cambiata descrizione parametro BaudrateReader al §4.10
- Aggiunte note relative alle ulteriori modalità di esenzione da verifica biometrica ai §10.2 e 11.1
- Modificata procedura di enrollment al §11.1 per l’inserimento manuale del codice
- Cambiate 3 schermate e modificata voce “no verifica da tessera” al §11.1
- Aggiunte 2 schermate e nota relative alla possibilità di saltare la scansione al §11.1
- Cambiata descrizione e aggiunta nota sul reset di fabbrica al §4.7
- Cambiata immagine e aggiunta descrizione checkbox “visitors free pass” e “enroll authorized” al §11
- Aggiornata lista con nuovi parametri FreePass, EnrollAuth, SkipBioVerify, DisableFunctions al §4.10
- Aggiunto nuovo paragrafo §11.4 relativo alle ulteriori modalità di esenzione da verifica biometrica
- Aggiunto nuovo formato a 71 byte e descrizione flag B al §5.4
- Aggiunta nota sul messaggio “utente non enrollato” al §10.2
- Aggiunta nota sul parametro EnrollAuth al §11.1
- Il flag ‘R’ è ora gestito al §5.4
- Aggiunta descrizione nuovi flag ‘R’ e ‘M’ ai file USERCODS.TXT e BIOUPDATE.TXT al §11.1
- Modificata descrizione e schermate al §4.8
- Aggiunto 2 lettori opzionali su NeoMAX e doppia scheda di espansione I/O al §2
- Aggiunte descrizione 2 lettori opzionali su NeoMAX al §3.6
- Modificati §3.3, 3.4 e 3.7 per la gestione della doppia scheda di espansione
- Aggiunti nuovi par. GateSensor1 e GateSensor2 e modificati valori di default par. GateState1 e GateState2 al §4.10
- Cambiato valore di default par. GateState1 e GateState2 al §6.1
- Aggiunta note sui lettori opzionali su NeoMAX ai §4.10, 5.4, 7, 11, 11.4, 15
- Aggiunta nota sullo schema di connessione della RS485 al §3.7
- Modificati §6, 6.1, 6.3 e 6.4 per il nuovo funzionamento degli input/relé con 2 schede NeoMAX
- Aggiunta nota relativa al parametro DisableFunctions per disabilitare il PIN ai §5.9 e 10.4
- Aggiornata lista con nuovi parametri RetryTimeout e ReviewDaysTA al §4.10
- Modificati casi relativi ai messaggi Tessera disabil.” e “Tessera non valida” al §5.13
- Aggiunte nuove codifiche HID/Wiegand 30bit, 37bit BCD e Corp. 1000-40 digits al parametro CardDecode al §4.10
- Aggiunta nota su parametro ReviewDaysTA al §10.7
- Aggiunto nuovo formato a 470 byte e descrizione flag d al §5.8
- Modificato casi relativi al messaggio “Timemod assente” al §5.13
- Aggiunti nuovi valori campo “CONTROLLI” al §7
- Aggiunti comandi per interrogare il modulo biometrico e cancellazione utenti al §12.4
- Cambiate schermate ai §4 e 4.11

Correzioni del documento dalla versione X1-X2 Manuale Utente r.1.6:

- Aggiunto valore “9999” del parametro ConnectionInterval al §4.10 e nota relativa al §15

- E’ sempre possibile inserire manualmente il codice nel menu biometrico, opzioni Enrollment, Cancella Utente e Amministratore al §11.1
- Aggiunta frase al parametro AllowTypeCode al §4.10
- Aggiunto parametro EnrollAll al §4.10 e descrizioni al §11, aggiunta nota al par. EnrollAuth al §4.10
- Cambiata schermata al §11
- Aggiunte opzioni di disattivazione e attivazione indefinita del relé al §4.2
- Aggiunto nuovo §16 per i comandi via FTP e cambiata numerazione §16 e §17 in §17 e §18
- Aggiunte note relativa all’esportazione e cancellazione biometrica via file FTP al §11.1
- Aggiunta nota relativa all’esportazione delle transazioni via file FTP al §7
- Piccola aggiunta al §4
- Aggiunta descrizione parametri ScreenOK e ScreenError al §4.10 e nota al §10.3
- Cambiati valori di default parametri TimeOutOpen e TimeOutClosed al §4.10 e §6.2
- Aggiunto nuovo parametro HideTypedCode al §4.10 e note ai §10.2 e §10.3
- Aggiunto nuovo parametro EnableNeoMaxI/O al §4.10 e note ai §3.7 e §4.8

Correzioni del documento dalla versione X1-X2 Manuale Utente r.1.7:

- Aggiunta nota al parametro Offline al §4.10
- Aggiunto tag relay=<indice>, <tempo> e nota al tag relay1=<tempo> al §11.2
- Corretti indici relé su scheda NeoMAX con indirizzo 2 al al §3.3
- Aggiunte note relative al passaggio all’ora legale/solare nella sezione TimeSettings al §4.10
- Aggiunto parametro MultiFormat al §4.10
- Modificata descrizione valore 26 del parametro CardDecode al §4.10
- Aggiunta nota sull’aggiornamento delle lingue al §8
- Cambiata la descrizione della schermata di selezione della verifica biometrica al §11.1
- Aggiunti nuovi parametri EnableHTTPServer, EnableFTPServer e MinimumQuality al §4.10
- Aggiunta descrizione numero di utenti, parametro MinimumQuality e cambiata descrizione di ImageQuality al §11
- Cambiate descrizioni al §11.1
- Aggiunta modalità auto scroll ai §10.6 e §10.7
- Aggiunto riferimento al programma Xatl@s al §2
- Aggiunta nota per scarico di un file su PC da menu HTTP al §4
- I lettori Legic adesso sono supportati al §2
- Aggiunti valori 36 e 38 del par.CardDecode al §4.10

Correzioni del documento dalla versione X1-X2 Manuale Utente r.1.8:

- Aggiunto supporto lettori Legic e descrizione comando di configurazione autoread al §4.7
- Aggiunti valore 37 del par.CardDecode e cambiata descrizione valori 36 e 38 al §4.10
- Aggiunto valore 0 al parametro BaudRateReader al §4.10
- Aggiornata lista sezioni al §4.9
- Invertito l’ordine delle sezioni [System] e [TimeSettings] al §4.10
- Cambiato valore di default dei parametri TimeoutOpenExtended, TimeoutClose, TimeoutCloseExtended al §4.10

- Aggiunto nuovo parametro ReaderLeds al §4.10
- Aggiunte note relative alla gestione dei LED al §3.6
- Cambiata frase al §4.3 per chiarificare meglio
- Aggiunti valori 78 e 79 del parametro CardDecode al §4.10

Correzioni del documento dalla versione X1-X2 Manuale Utente r.1.9:

- Aggiunto nuovo valore del campo "Dial number" al §15
- Aggiunto nuovo §11.5 e nota al §11.3 per il caso di sistema biometrico distribuito sotto Xatlas
- Aggiunto parametro LogAggressiveFlush al §4.10
- Inserite note sulla copia delle micro-SD card al §1
- Cambiata descrizione parametri FacilityCode e FacilityCodeBegin al §4.10
- Cambiata immagine scheda e descrizione pinatura al §3.2
- Aggiunta sezione [Printer] al §4.10
- Aggiunta porta seriale TTL o RS232 al §2
- Aggiunto parametri MandatoryFunction e EnableFastMenu al §4.10
- Cambiato valore di default del par.Contrast al §4.10
- Aggiunto nuovo §3.8 per selezione livelli della porta seriale e cambiata numerazione del vecchio §3.8 in §3.9
- Aggiunto nuovo §4.8 per I file della stampante e incrementata numerazione dei vecchi §4.8, §4.9, §4.10 e §4.11
- Cambiato nome paragrafo e aggiunti nuovi formati record di FKEY.TXT e descrizione file ENQUIRY.TXT §4.5

Correzioni del documento dalla versione X1-X2 Manuale Utente r.2.0:

- Aggiunta mappa dei caratteri russo / cirillico e cambiata modalità selezione set caratteri al §18
- Aggiunti valori 41, 42 e 43 del par.CardDecode e cambiata descrizione valori 30, 32, 36, 37, 38 al §4.11
- Aggiunto modulo fw opzionale RFID2 Serial Zucchetti al §4.12
- Aggiunti parametri DeniedRelay, DeniedRelayTimeout e Indexing al §4.11
- Completa revisione della sezione [AccessControl] al §4.11, inclusi tutti i parametri relativi a input e relé
- Aggiunta del varco di tipo "centrale allarme" al §6.1 e al par. GateType al §4.11
- Cambiata descrizione parametri SecurityBootAuth al §6.3
- Cambiata descrizione dei §4.10 e 4.12 relativa alle opzioni "Reset default parameters" e "Format SD Card"

Correzioni del documento dalla versione X1-X2 Manuale Utente r.2.1:

- Diverse correzioni a sintassi, numerazione paragrafi e riferimenti incrociati
- Aggiunta mappa dei caratteri greco
- Aggiornate tutte le schermate dell'interfaccia web
- Cambiata dimensione micro-SD a 2GB ai §1 e §2
- Aggiunta nota relative al par. ReaderLeds e cambiata descrizione della procedura di test dei lettori al §3.6
- Aggiunti standard barcode supportati al §2
- Aggiunto supporto per le connessioni/disconnessioni GPRS schedulate ai §4.2 e §15
- Aggiunti parametri, GateSensor1, GateSensor2, PendingAlarms, EditionBegin, EditionLength, SaveReaderSource, SendTemplate, VirtualKeyIn1, VirtualKeyIn2, HttpPort, ResponseTimeout al §4.11
- Cambiata descrizione del par. ConnectionInterval al §4.11

- Aggiunte note relative all'uso di un modulo Legic R&W e descrizione dei formati di decodifica Legic al §4.7
- Leggermente cambiata descrizione al §4.4
- Aggiunte note relative agli stati degli ingressi digitali al §4.5
- Cambiata descrizione configurazione via file TXT al §4.6
- Cambiata descrizione del par. ReaderLeds al §4.11
- Il par. FontEncoding non è più usato al §4.11
- Cambiata descrizione del campo R e aggiunto campo 'ee' per il controllo edizione al 5.4
- Aggiunto errore "Edizione non valida" al §5.11
- Aggiunto nuovo §5.14 relativo all'editor web per le tabelle di controllo accessi
- Corretti valori di default dei tempi consentiti per il passaggio al §6.2
- Aggiunto par. DeniedRelay al §6.4
- Aggiunto campo EDIZIONE e cambiata descrizione dei valori 34 e 51 del campo CONTROLLI al §7
- Aggiunta nota relativa alla connessione da client FTP al §7.3
- Le stringhe in portoghese non sono più presenti per default al §8
- Aggiunte checkbox Save Reader Source e Send Template to server al §11
- Cambiato note relative all'opzione Erase Mifare Card, cambiata descrizione del campo R nei record di USERCODS.TXT e aggiunta nota relativa al par. SendTemplate al §11.1
- Cambiati valori del par. CardDecode al §11.2
- Leggermente cambiata descrizione e aggiunta distribuzione online dei template al §12
- Cambiata la descrizione del tag $transaction$ al §12.1
- Aggiunto campo "print=" e rimossa vecchia nota2 per il campo "pin=" al §12.2
- Aggiunto campo "gprs=off" e comandi RESTART e BIOADD al §12.4
- Corretti valori di esempio del par. MasterURL ai §12.1 e §12.5
- Aggiunto nuovo §12.6 per la distribuzione online dei template
- Aggiunto server tag io al §12.1
- Rimosse frasi erronee ai §4.2, 4.3, 4.4, 4.5, 4.6
- Aggiunto menu "Info/GPRS" al §10.5

Correzioni del documento dalla versione X1-X2 Manuale Utente r.2.2:

- Rimosso campo errato '00' al §5.4
- Aggiunto terminatore fisso "_0", e l'identificatore utente è ora gestito nei template online al §12.6
- Cambiato il formato e la descrizione del comando BIOADD al §12.4
- Cambiato indirizzo IP di default quando il server DHCP non risponde al §3.5
- Cambiata descrizione del parametro IPAddress al §4.11
- La lunghezza max del tag $fullcode$ adesso è di 80 caratteri al §12.1
- Agginute note relative alla sostituzione dei caratteri spazio ai §12.1 e §12.4
- Aggiunto comando RDR al §12.4
- Aggiunteanota sulla gestione dei campi comando nelle risposte alle transazioni online al §12.4
- Cambiati tutti i riferimenti a RFID2 in RFID2/3

- Aggiunti menu di attivazione dei server HTTP e FTP al §10.5
- Aggiunto lettore RS232 ausiliario su connettore a vite al §2
- Aggiornate tutte le schermate al §4
- Aggiornate tutte le schermate al §5.14
- Adesso tutti i lettori possono essere usati per inserire dei codici tessera nel web editor al §5.14
- Cambiata nota relativea all'ora legalenella sezione [TimeSettings] al §4.11
- Aggiunta identificazione tramite colorazione dei campi al §5
- Il valore '34' del campo CONTROLLI non è più usato al §7
- Aggiunto nuovo §17 per la versione speciale Aperio e cambiata numerazione dei capitoli §17 e §18 in §18 e §19
- Aggiunta nota relativa alla porta del server HTTP sull'host al §12.1
- Cambiata descrizione dei parametri HttpPort, FtpPort, MasterURL e ServerURL al §4.11
- Aggiunto valore '91' del parametro CardDecode al §4.11
- Cambiata leggermente la descrizione del parametro UTC al §4.11
- Aggiunta sezione [Aperio] al §4.11
- Aggiunta sequenza Ctrl+F5 per svuotare la cache del browser al §4
- Aggiunto valore 86 del parametro CardDecode al §4.11
- Aggiunti riferimenti ai file di definizione del formato degli scontrini da stampare per le enquiries locali al §4.5

Correzioni del documento dalla versione X1-X2 Manuale Utente r.2.3:

- Tradotti campi EDITION e DAYLIGHT a rimosso campo RFU al §7
- Aggiunti segnaposto %d e %h al file PRINTER.TX al §4.8
- Cambiato link alla partners area al §18
- Aggiunta avvertenza relativa al PIN riservato "9999" al §5.9
- Aggiunta nota relativa al caricamento del fw via http con tag "file=" in risposta al keepalive, al §12.4
- Aggiunti parametri "DirectionInput", "BackLight", "TurnoffEthernet", "TTY1Busy" al §4.11
- Aggiunti valori 27,28,47,65,66,67,68,69,70,76,77,99 al par. CardDecode al §4.11
- Aggiornata descrizione diversi valori del par. CardDecode al §4.11
- Aggiunti comandi Legic per lo UID in formato standard o con byte scambiati al §4.7
- Aggiunta descrizione comandi CR A… per i nuovi lettori RFID4 Legic con chipset SM-4200 al §4.7
- Aggiunta opzione '!' per il riavvio automatico da file ALARMS.TXT al §4.2
- Aggiunti segnaposti 'E' e 'ee' per il formato custom delle transazioni al §7.1
- Aggiunta nota a proposito del Web Table Editor ai §4.2 e §4.3
- La chiave fw GATE MANAGER non è più necessaria ai §4.12 e §6
- Cambiate diverse immagini e descrizioni al §5.14
- Aggiunta nota relativa alla pulizia della cache del browser quando si aggiorna il Web Table Editor al §5.14
- Cambiata descrizione del parametro RecordInvalidAccess al §4.11
- Aggiunto stato batteria nel menu Info/Power al §10.5
- Cambiata descrizione del §4.5 e del parametro MandatoryFunction al §4.11
- Rimosso il menu a discesa "Lightning condition" al §11

- Il parametron Lightning Condition è ora fisso a '0' al §4.11
- Aggiornato tutto il §17 (ma solo in inglese)

Correzioni del documento dalla versione X1-X2 Manuale Utente r.2.4:

- Modificata dimensione microSD ai §1 e 2
- Cambiate caratteristiche tecniche al §2
- Tolti tutti i riferimenti a XAM
- Aggiunti riferimenti alle versioni Black
- Aggiunto montaggio X1/X2 Black al §3.1
- Cambiate decodifiche 32 bit da (12+18bit) a (13+17bit) al §4.11
- Aggiunta nota su accesso al menu supervisore se è stato definito un utente amministratore ai §10.5 e 11.1
- Aggiunte codifiche 74 e 75 e modificata descrizione valori 13, 15, 19 e 57 al par. CardDecode al §4.11
- Aggiunta necessità di formattazione FAT32 al §14
- Cambiata descrizione homepage server http, aggiunto account manager al §4
- Aggiornate schermate ai §4, 5.14, 7.3, 8, 11, 15, 19
- Aggiunti spazi di riempimento ai contatori di presenza al §4.8
- Cambiato valore di default dei par. CardDecode al §4.11
- Cambiato par. BufferSize in MaxPendingRecord e cambiata descrizione ai §4.11 e §7
- Cambiata descrizione par. DeleteOld al §4.11
- Aggiunti par. ManagerPassord, UseNTP, NTPServerName e NTPRefresh al §4.11
- Riaggiunti parametri rimossi per sbaglio nella sezione [USB] al §4.11
- Cambiata descrizione parametri AutoDaylightSavingTime e UTC al §4.11
- Aggiunta sincronizzazione SNTP al §4.1
- Cambiata descrizione del par. LogLevel e aggiunta nota al par. TTY1Config al §4.11
- Aggiornata descrizione CLOKI al §5.14
- Aggiunta nota e cambiata descrizione al §6
- Invertita numerazione e cambiate alcune descrizioni ai §7.1 e 7.2
- Aggiunto §9.1 per l'aggiornamento FW dei lettori
- Cambiato max numero causali al §4.4
- Cambiato link area partner al §18
- Aggiunte note varie al §15.1
- Aggiunta nota sulla revisione dati quando c'è il file ENQUIRY.TX ai §4.5 e 10.7
- Aggiunte note relative a CLOKI ai §4.4 e 4.5

Correzioni del documento dalla versione X1-X2 Manuale Utente r.2.5:

- Aggiunto supporto ai file PRINTER_Rcc..cc.TXT ai §3.8 e 4.8
- Cambiati tutti gli screenshot del web server
- Cambiata descrizione account amministratore e CLOKI (ora si possono cambiare anche gli username) al §4
- Estesa descrizione par. ReviewDaysTA ai §4.11 e 10.7
- Modificata politica di blocco delle transazioni al §7

- Cambiata descrizione parametri MaxPendingRecord e DeleteOld al §4.11
- Cambiate descrizione errore “Tessera non valida” al §5.16
- Aggiunti nuovi campi vuoti nel record di TRANSACTIONS.TXT e descrizioni al §7
- Cambiato formato del record di evento e aggiunti valori al §7.1
- Aggiunti nuovi segnaposto per record custom al §7.2
- Aggiunta nota sui diversi formati del file USERS.TXT al §5.9
- Aggiunto nuovo §17 sulla crittografia e cambiata numerazione dei capitoli già esistenti 17-19 in 19-20
- Aggiunti nuovi parametri Payload, CardLayoutLength, CryptoEnabled, TrnsHash, EncodeUrl al §4.11
- Cambiati nomi e descrizioni dei parametri con le passowrd criptate al §4.11
- Aggiunta nota relativa al comando http RDR al §12.4
- Aggiunti nuovi flag nei record e comandi biometrici ai §11.1, 12.4, 17.2.5
- Cambiato formato dei messaggi online di tipo bioupdate al §12.6
- Rimosso codice IMEI dai menu di configurazione GPRS ai §10.5 e §15
- Aggiunta specifica dei codici causali puramente numerici ai §4.1. e 5.10
- Aggiunta nota su export .csv al §5.14

Correzioni del documento dalla versione X1-X2 Manuale Utente r.2.6:

- Aggiunta nota su nome file PRINTER_Rcc..cc.TXT se si utilizza AXREASON.TXT al §4.8
- Aggiunta opzione causali numeriche “libere” ai §4.5 e §10.6
- Aggunto parametro AllowTypeReason al §4.11

Zucchetti Axess Spa
Via Lepetit, 40
20020 Lainate (MI) - Italy
Tel: +39 0371 594 7000
Fax: +39 0371 594 7170
Web: www.axesstmc.com
Email contact@axesstmc.com
support@axesstmc.com

Via della Filanda, 22
40133 Bologna - Italy
Tel: +39 0371 594 7311
Fax: +39 0371 594 7399

Zucchetti Axess USA
1600 Osgood Street
Suite 2056
North Andover, MA 01845. USA
Tel: +1 978 258 9522