

ZP1 & ZP2



Manuale Utente

(r.1.8 - Marzo 2013 – compatibile con le versioni di FW “a10 build 93” e successive)

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

SOMMARIO

1. INTRODUZIONE.....	4
2. CARATTERISTICHE TECNICHE.....	5
3. INSTALLAZIONE.....	7
3.1 APERTURA E MONTAGGIO	7
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 VERSIONE HARDWARE	12
4. CONFIGURAZIONE	13
4.1 IMPOSTAZIONE DATA E ORA	16
4.2 ALARMS.TXT	17
4.3 HOLIDAYS.TXT.....	17
4.4 REASONS.TXT.....	17
4.5 FKEY.TXT	18
4.6 DIRECTION.TXT	18
4.7 READER1.TXT, READER2.TXT, EXTREADER.TXT	19
4.8 CONTROLLO REMOTO DEI RELE' E DEGLI INGRESSI DIGITALI DA WEB SERVER http.....	20
4.9 IMPOSTAZIONE PARAMETRI.....	22
4.10 LISTA DEI PARAMETRI.....	23
4.11 ATTIVAZIONE DI FUNZIONI OPZIONALI DEL FIRMWARE	50
5. TABELLE DI CONTROLLO ACCESSI	52
5.1 FILE NECESSARI PER IL CONTROLLO ACCESSI.....	52
5.2 FILE OPZIONALI PER IL CONTROLLO ACCESSI	53
5.3 FORMATO DEI FILE PER IL CONTROLLO ACCESSI.....	54
5.4 CARDS.TXT.....	55
5.5 CARDRNGE.TXT	57
5.6 AUTHGRP.TXT.....	58
5.7 AUTH.TXT	59
5.8 TIMEMOD.TXT	60
5.9 USERS.TXT	61
5.10 AXREASON.TXT	63
5.11 CALENDAR.TXT.....	65
5.12 LA MODALITA' "SOLO PIN"	65
5.13 MESSAGGI DI ERRORE	66
6. GESTIONE DI UN VARCO	67
6.1 TIPO DI VARCO	67
6.2 TEMPI MASSIMI CONSENTITI PER IL PASSAGGIO	68
6.3 ASSEGNAZIONE DEGLI INGRESSI DIGITALI	69

6.4 ASSEGNAZIONE DELLE USCITE RELE'	71
6.5 GESTIONE ONLINE DEL VARCO.....	73
7. TRANSAZIONI.....	75
7.1 DEFINIZIONE DI UN FORMATO PERSONALIZZATO	77
7.2 EMISSIONE E RIENTRO DI EVENTI RELATIVI ALLA GESTIONE DI UN VARCO	78
7.3 INVIO DELLE TRANSAZIONI TRAMITE CLIENT FTP	79
8. LINGUE	80
9. AGGIORNAMENTO FIRMWARE	81
10. INTERFACCIA UTENTE DI ZP1/ZP2	82
10.1 AVVIO	82
10.2 STATO DI ATTESA (PRONTO AD ACCETTARE TRANSAZIONI)	83
10.3 DOPO UNA LETTURA DI CARTA, DIGITAZIONE DI CODICE O AUTENTICAZIONE BIOMETRICA	86
10.4 RICHIESTA CODICE PIN	87
10.5 MENU SUPERVISORE	88
10.6 TRANSAZIONI CON CODICE CAUSALE.....	91
10.7 REVISIONE DATI DI PRESENZA	92
10.8 MENU "RIDOTTO" PER SELEZIONE CAUSALI / ENQUIRIES REMOTE.....	93
11. IL MODULO BIOMETRICO ESTERNO FINGERBOX	94
11.1 MENU DI GESTIONE DELL'ARCHIVIO DELLE IMPRONTE	98
11.2 SALVATAGGIO DELLE IMPRONTE SU CARTE MIFARE	107
11.3 IMPORTAZIONE DI IMPRONTE NEL MODULO BIOMETRICO VIA FTP	109
11.4 ULTERIORI MODALITA' DI ESENZIONE DALLA VERIFICA BIOMETRICA.....	111
12. TRANSAZIONI ONLINE VIA HTTP.....	111
12.1 MESSAGGI HTTP PER TRANSAZIONI ONLINE (DA ZP1/ZP2 A MasterURL).....	111
12.2 FORMATO RISPOSTA DEL SERVER (DA MasterURL A ZP1/ZP2)	113
12.3 MESSAGGIO "KEEP ALIVE" (DA ZP1/ZP2 A MasterURL)	114
12.4 FORMATO RISPOSTA DEL SERVER AL "KEEP ALIVE" (DA MasterURL A ZP1/ZP2).....	114
12.5 MODALITA' ONLINE: SERVER NON IN LINEA	118
13. FUNZIONAMENTO A BATTERIA	119
13.1 RICARICA RAPIDA DELLA BATTERIA.....	119
14. UTILIZZO DELLA CHIAVETTA USB	120
14.1 SCARICO TRANSAZIONI SU CHIAVETTA USB.....	120
14.2 SALVATAGGIO CONFIGURAZIONE SU CHIAVETTA USB	121
14.3 CARICAMENTO CONFIGURAZIONE DA CHIAVETTA USB.....	121
14.4 AGGIORNAMENTO FIRMWARE DA CHIAVETTA USB.....	122
14.5 SALVATAGGIO DATI BIOMETRICI SU CHIAVETTA USB.....	122
14.6 IMPORTAZIONE DATI BIOMETRICI DA CHIAVETTA USB	122
15. USO DEL MODEM GPRS OPZIONALE.....	124
15.1 VISUALIZZAZIONE STATO MODEM GPRS	127
16. ESECUZIONE DI COMANDI VIA FTP	128
17. STRUMENTI SOFTWARE	128
18. MAPPE DEI CARATTERI	129

1. INTRODUZIONE

Per chi già conosce i terminali della linea ZUCCHETTI, ZP1 e ZP2 sono differenti sia per come vengono configurati che per come comunicano.

Con ZP1 e ZP2 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 ZP1 e ZP2, con una sola eccezione per consentire una facile identificazione di tutti i terminali ZP1/ZP2 in rete (vedi §3.5 a pag. [10](#))

Su ZP1 e ZP2 il file system si trova su una micro-SD card rimovibile.

La capacità della micro-SD card è almeno 1 GB: ciò significa che è possibile registrare sulla memoria del terminale un numero enorme di transazioni e utenti autorizzati, tanto da renderne superfluo il calcolo.

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 ZP1/ZP2, e l'applicazione host non si accorgerà neppure che il terminale è stato sostituito: l'unica cosa che cambia è l'indirizzo MAC.

ZP1 e ZP2 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.

2. CARATTERISTICHE TECNICHE

Tastiera	<ul style="list-style-type: none"> 6 tasti funzione a membrana ai lati del display <u>Solo X2</u>: 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 1GB su micro-SD card interna. Nota: l'alloggiamento della micro-SD è accessibile solo aprendo il terminale, cioè rimuovendo la cornice frontale e le 4 viti agli angoli
Lettore Primario	RFID integrato (Clk&Data 125KHz o HID, Mifare o Legic seriale) o esterno magnetico Tk2 o Tk1/2/3 o barcode; è anche possibile gestire un lettore generico esterno con interfaccia Wiegand
Lettori Ausiliari	<ul style="list-style-type: none"> Esterno (RFID Clk&Data 125KHz o HID / Mifare o Legic seriale, Magnetico Tk2 o Tk1/2/3, barcode) oppure, <u>in alternativa</u>, 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. Utilizzabile solo <u>in alternativa al modem GPRS opzionale</u> 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	<ul style="list-style-type: none"> Protocolli di comunicazione: TCP/IP, HTTP e FTP Ethernet : 10/100 Mb/s PoE Uscita Wiegand 37bit H10302 su connettore a vite estraibile per la ritrasmissione dei codici ricevuti dei lettori in locale (<u>in alternativa alla gestione dei 2 LED di un lettore esterno</u>) 1 porta USB HOST 2.0 ad alta velocità per chiavette di memoria Disponibili versioni con modem GPRS opzionale (da richiedere all'acquisto del terminale), utilizzabile solo <u>in alternativa al lettore ausiliario esterno su connettore a vite</u>
Input/Output	<ul style="list-style-type: none"> 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), utilizzabili esclusivamente per la logica di gestione del varco 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 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 i <i>middleware</i> Traxit32 e XAM e con il programma di controllo accessi Xatl@s
Caratteristiche Fisiche	Grado di protezione ambientale: IP55 Contenitore: ABS V0 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

Per aprire ZP1 e ZP2 occorre prima rimuovere la cornice frontale, facendo leva sulle rientranze lungo i bordi superiore e inferiore (vedi vista dall'alto, figura 1).

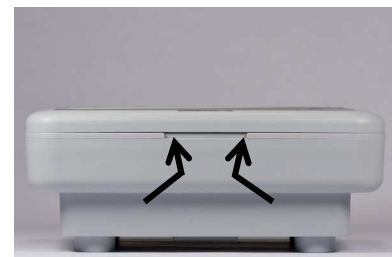


Figura 1

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.

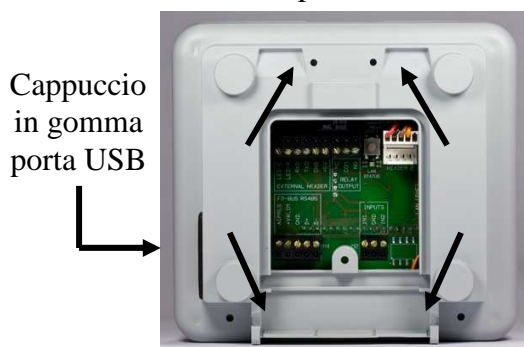
Supporti da forare per
montaggio diretto a muro



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

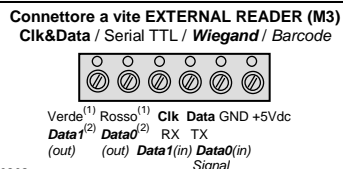
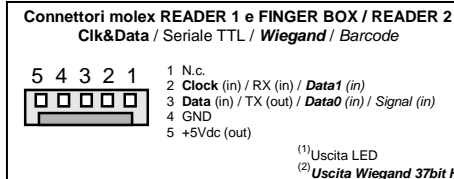
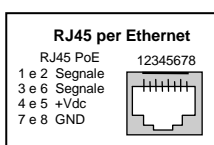
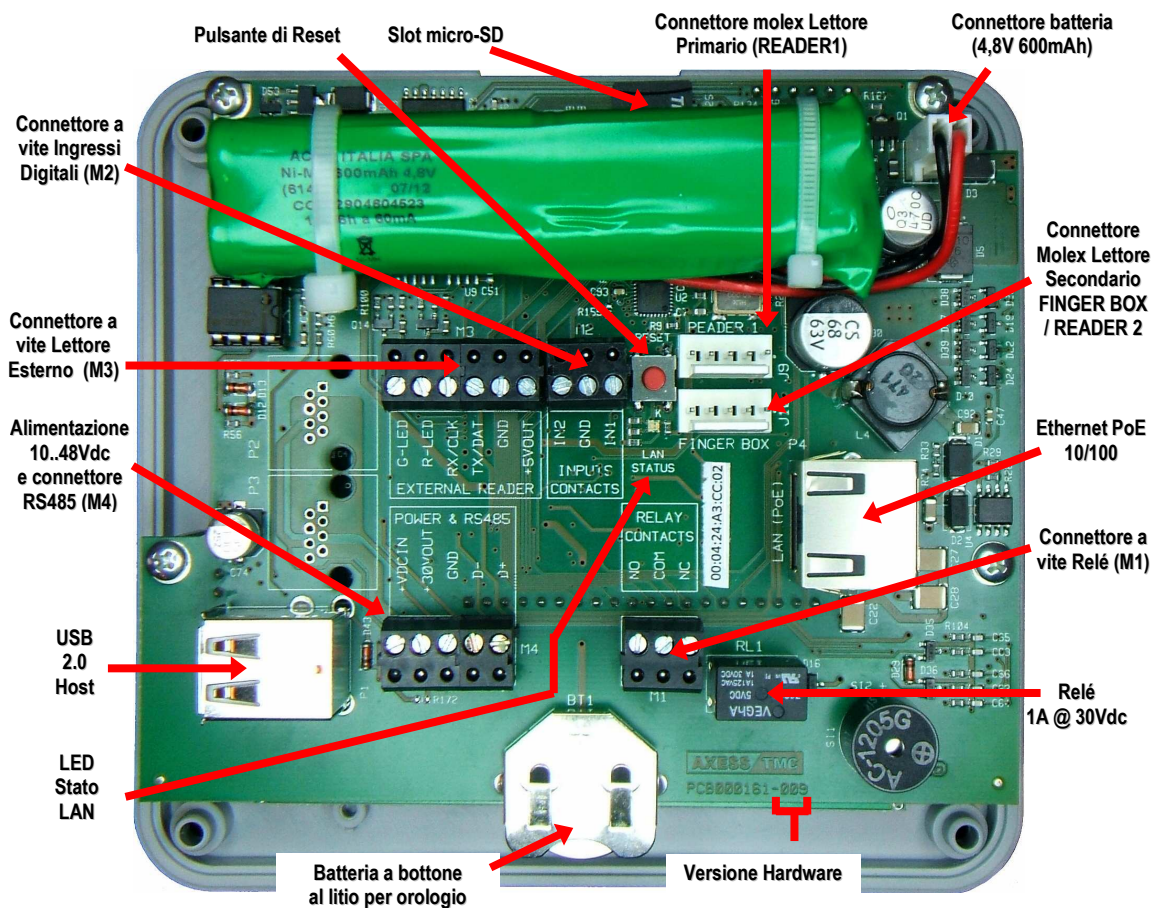
Il vantaggio principale di usare una staffa a muro è che si può facilmente rimuovere l'intero terminale dal muro e accedere a tutte le connessioni esterne attraverso la finestra posteriore, senza bisogno di aprire il contenitore.

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 §3.8 a pag. 12).

3.2 ALIMENTAZIONE, BATTERIE E COLLEGAMENTI PRINCIPALI

ZP1 e ZP2 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 **009**.



Attenzione: ZP1 e ZP2 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: una ricarica rapida completa (disponibile solo a partire dalla versione di hardware **006**, vedi §3.8 a pag. [12](#) e §13.1 a pag. [119](#)) 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 ZP1/ZP2 debbano essere installati in ambienti ove la temperatura ambiente può superare i 40°C, si consiglia di posizionare le batterie di ZP1/ZP2 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 RELÉ

ZP1/ZP2 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. [8](#)).

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 ZP1/ZP2, 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 ZP1/ZP2 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

ZP1/ZP2 dispone di 2 ingressi digitali per contatti puliti utilizzabili esclusivamente per la logica di gestione del varco, vedi §6 a pag. 67, e che quindi non possono essere effettuati 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. 8) 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 ZP1/ZP2 possono essere usate per contatti puliti. Per la posizione dei contatti degli ingressi digitali sulla scheda 914 NeoMAX fare riferimento al pieghevole allegato alla scheda stessa. Per dettagli sulla connessione fra ZP1/ZP2 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. 8). Se lampeggia, significa che è stata rilevata attività di rete.

ZP1 e ZP2 vengono consegnati con DHCP abilitato, ma se il server DHCP non risponde, il terminale assume l'indirizzo IP di default 192.168.1.240. In ogni caso, l'indirizzo MAC e le impostazioni IP correnti vengono mostrate all'accensione (vedi §10.1 a pag. 82), e possono essere facilmente modificate entrando nel menu supervisore (§10.5 a pag. 88).

E' anche possibile identificare facilmente da remoto tutti i terminali ZP1/ZP2 in rete, 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 ZUCCHETTI SPA, inclusi X1 and X2, verranno trovati e identificati:

- X** → ZP1/ZP2 risponde inviando la configurazione IP corrente nel formato standard compatibile con EtherLite, con la prima linea che riporta parzialmente la versione fw nel formato "**VNNx**"
- h** → ZP1/ZP2 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 §4.11 a pag. 50)
- V** → si tratta di un nuovo comando TMC-UDP non supportato da tutti gli altri terminali della linea ZUCCHETTI: ZP1/ZP2 risponde inviando la versione fw estesa e la data/ora del terminale nel formato "**X1 aNN build nnn, MMM gg aaa 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 §4.10 a pag. 42) viene impostato ad un indirizzo IP valido e raggiungibile ZP1/ZP2 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, ZP1/ZP2 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. 114: 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 §4.10 a pag. 42) viene impostato ad 1 (default 0).

3.6 COLLEGAMENTO LETTORI

Fino a 3 lettori di carte diversi possono essere collegati direttamente a ZP1/ZP2 (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. 8). 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 §3.8 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 contrassegnata come "EXTERNAL READER", ma solo nel caso in cui non sia già presente il modem GPRS opzionale (vedi §15 a pag. 124). 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"). 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 §4.10, pag. 37).

In tutti i casi si raccomanda di seguire attentamente l'ordine dei pin riportato nello schema a pag. 8, 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 ZP1/ZP2 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 ZP1/ZP2 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 §4.10 a pag. 33) ad un valore compatibile con lo specifico formato di uscita dei dati di ciascun tipo di lettore. 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. 13; 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 §4.10 a pag. 37) 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 "**Reader used by finger box**" e la posizione del connettore corrispondente. Per maggiori informazioni si veda il §11 a pag. 94.

(*) Come detto la sezione *[ExtReader]* si applica anche ad entrambi i lettori aggiuntivi su schede 914 NeoMAX, 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.

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 ZP1/ZP2 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.

ZP1/ZP2 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. 8; 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 ZP1/ZP2 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 ZP1/ZP2, collegandole anche al morsetto **+VDCIN** del connettore a vite estraibile **M4**. Se invece ZP1/ZP2 è alimentato in **PoE**, le schede 914 NeoMAX possono comunque essere alimentate direttamente da ZP1/ZP2 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. 29) deve essere impostato a '1' (default).

3.8 VERSIONE HARDWARE

La versione hardware di ZP1/ZP2 è serigrafata sul lato inferiore della scheda a circuito stampato (vedi figura a pag. 8, 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. 119) e la gestione della porta USB host (vedi §14 a pag. 120) 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. 120) 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) Mediante XAM, il nuovo *middleware* disponibile per lo scarico nell'area partners della Zucchetti SPA
- 2) Caricando file di testo .TXT con qualunque programma client FTP (eccetto FireFox FireFTP) nella memoria del terminale
- 3) Collegandosi alla pagina iniziale del web server HTTP del terminale con qualunque browser standard
- 4) Con un programma client HTTP che invia opportuni comandi in risposta ai messaggi "Keep Alive" ricevuti dal terminale
- 5) Solo per alcune impostazioni: direttamente dal terminale (menu supervisore), vedi §10.5 a pag. [88](#)
 - Il metodo 1 è spiegato in dettaglio nella "XAM User Guide".
 - Il metodo 2 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.
 - Il metodo 3 è il più intuitivo dal punto di vista dell'utente, poiché consente di configurare il terminale mediante un menu a interfaccia grafica disponibile alla pagina iniziale del web server HTTP del terminale (http://<Indirizzo_IP_Terminale>) vedi figura in basso.

Nota: l'unico nome utente riconosciuto dal sistema è "admin", e la password assegnata inizialmente all'utente "admin" è ancora "admin". Una volta avuto accesso al menu HTTP con queste credenziali, potrete modificare la password (la modifica viene applicata sia per gli accessi in FTP che per quelli in HTTP).

X1/X2 Configuration

Network	Network
GPRS modem	Terminal ID <input type="text" value="01"/>
FTP Client	MAC address <input type="text" value="00:04:24:A3:5C:28 [CA,F1]"/>
Time & Attendance	DHCP <input checked="" type="checkbox"/>
Access Control	IP Address <input type="text" value="192"/> . <input type="text" value="168"/> . <input type="text" value="1"/> . <input type="text" value="171"/>
Reader 1	Subnet Mask <input type="text" value="255"/> . <input type="text" value="255"/> . <input type="text" value="255"/> . <input type="text" value="0"/>
Reader 2	Gateway <input type="text" value="192"/> . <input type="text" value="168"/> . <input type="text" value="1"/> . <input type="text" value="254"/>
External Reader	Primary DNS <input type="text" value="192"/> . <input type="text" value="168"/> . <input type="text" value="1"/> . <input type="text" value="1"/>
Daylight Saving Time	Secondary DNS <input type="text" value="192"/> . <input type="text" value="168"/> . <input type="text" value="1"/> . <input type="text" value="2"/>
Time and Date	Master URL <input type="text"/>
USB	Protocol <input checked="" type="radio"/> XAtI@s / FTP
System	<input type="radio"/> HTTP
Remote Relays	FTP Port <input type="text" value="21"/>
Biometrics	Connection Timeout <input type="text" value="5"/> seconds
File Manager	Keep Alive Interval <input type="text" value="15"/> seconds
Password	Retry connection timeout <input type="text" value="60"/> seconds
Log	<input type="button" value="Save"/>

HTTP Messages	
Online Message	<input type="text" value="/online?trsn=\$transaction\$&id=\$termid\$&mac=\$mac\$"/> <input type="button" value="Modify"/>
Batch Message	<input type="text" value="/batch?trsn=\$transaction\$&id=\$termid\$&mac=\$mac\$"/> <input type="button" value="Modify"/>
Keep Alive Message	<input type="text" value="/keepalive?id=\$termid\$&mac=\$mac\$&date=\$date\$&time=\$time\$&localtrsn=\$localtransa"/> <input type="button" value="Modify"/>
IP address	<input type="text"/> <input type="button" value="Ping"/>

Il web server HTTP, inoltre, consente di inglobare anche il metodo 2 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 "Sfogli..." 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):

X1/X2 Configuration

- Network
- Time & Attendance
- Access Control
- Reader 1
- Reader 2
- External Reader
- Daylight Saving Time
- Time and Date
- USB
- System
- Remote Relays
- Biometric
- File Manager
- Password
- Log

File Manager

Current directory: \

Upload File:

Create Directory:

Browse directory:

File Name	File Size	Creation Date	Delete
LANGUAGE.TXT	3162	15.09.2011 - 13:09	<input type="checkbox"/>
LOG.TXT	9534	16.09.2011 - 10:43	<input type="checkbox"/>
BATTLOG.TXT	7160	16.09.2011 - 11:23	<input type="checkbox"/>
BATTERY.TXT	26	15.09.2011 - 16:54	<input type="checkbox"/>
btransaction.loc	1040	16.09.2011 - 10:42	<input type="checkbox"/>
TRANSACTIONS.TXT	424	16.09.2011 - 10:42	<input type="checkbox"/>
PARAMETERS.TXT	1639	15.09.2011 - 18:12	<input type="checkbox"/>
FLASHCUR.BIN	317644	01.01.2010 - 12:00	<input type="checkbox"/>
BIOEXP	DIR	11.01.2012 - 17:40	<input type="checkbox"/>
BIOIMP	DIR	11.01.2012 - 17:40	<input type="checkbox"/>

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 il pulsante "aggiorna" del browser (o "F5" sulla tastiera) per visualizzare i dati aggiornati.

- Il metodo 4 è un modo alternativo al 2 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. 111) 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 §4.10 a pag. 42) oppure, analogamente, selezionando il **radio button "HTTP"** fra le opzioni "**Protocol**" nella pagina "**Network**" del web server HTTP (vedi figura a pag. 14): questa modalità standard di comunicazione sostituisce il protocollo proprietario TMC-UDP utilizzato sugli altri nostri terminali.

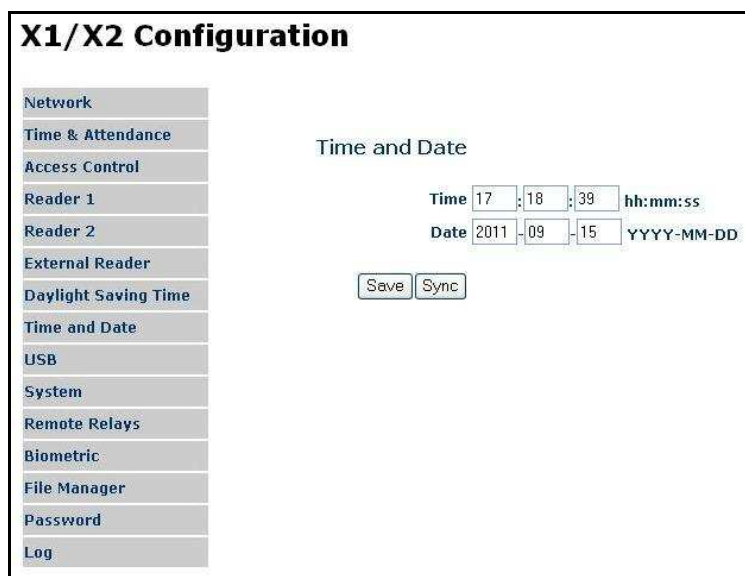
Il valore di default del parametro **Protocol** (0) corrisponde invece al **radio button "Xat@s / FTP"**: viene infatti usato nel caso in cui ZP1/ZP2 venga gestito da XAM (vedi punto 1) o dal programma Xat@s, oppure nel caso si intenda usare solo il protocollo FTP per configurarlo o scaricare in modalità *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. 114 per approfondire il concetto di messaggio "Keep Alive" via HTTP, ed il §12.4 a pag. 114 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 ZP1/ZP2, fate click sul link **"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:



X1/X2 Configuration

Network

Time & Attendance

Access Control

Reader 1

Reader 2

External Reader

Daylight Saving Time

Time and Date

USB

System

Remote Relays

Biometric

File Manager

Password

Log

Time and Date

Time 17 : 18 : 39 hh:mm:ss

Date 2011 - 09 - 15 YYYY-MM-DD

Save Sync

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. 81.

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 ZP1/ZP2 di cui viene specificato l'indirizzo IP.

Metodo manuale da menu supervisore

Nel caso in cui ZP1/ZP2 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. 88.

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. 79. Questo file può essere caricato solo via FTP, non si possono fare impostazioni via web server HTTP.

Il formato di ogni record è il seguente:

HHMM,R,TT,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 la funzionalità client FTP.

TT

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 la funzionalità client FTP, questo campo deve essere impostato al valore fisso "1" per consentire l'invio delle transazioni

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

4.3 HOLIDAYS.TXT

Lista delle date dei giorni festivi da considerare quando si usa ALARMS.TXT. Questo file può essere caricato solo via FTP, non si possono fare impostazioni via web server HTTP.

Il formato di ogni record è il seguente:

GGMM

Dove: GG=giorno del mese, MM=mese

4.4 REASONS.TXT

Lista dei codici causali e loro descrizioni. Questo file può essere caricato solo via FTP, non si possono fare impostazioni via web server HTTP. Sono consentite fino a 32 causali diverse.

Il formato di ogni record è il seguente:

CC..CC,descrizione

Dove CC..CC è il codice causale (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. 52). Si tenga presente che il file **AXREASON.TXT** (vedi §5.10 a pag. 63) 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).

4.5 FKEY.TXT

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. [91](#) (solo sui modelli X2 con tastiera numerica). Questo file può essere caricato solo via FTP, non si possono fare impostazioni via web server HTTP. Si possono definire fino a 10 tasti di scelta rapida diversi (tanti quanti sono i tasti numerici sulla tastiera dell'X2). Solo se ZP1/ZP2 viene gestito dal programma Xatl@s, è anche possibile associare alcuni tasti numerici alla scelta rapida di particolari *enquiries* remote invece che alla scelta rapida di causali: in questo caso, comunque, il file FKEY.TXT viene caricato automaticamente e non deve essere modificato. In ogni caso, le descrizioni delle sole causali e/o delle *enquiries* remote 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. [93](#). Ecco perché il file FKEY.TXT può comunque essere utilizzato anche sui modelli X1 che non dispongono della tastiera numerica.

Il formato del file è il seguente:

[Functions]

F_n=Rcc..cc

...

La prima riga **[Functions]** è fissa, e ad essa seguono uno o più record (max 10) aventi lo stesso formato, dove i caratteri mostrati in grassetto '**F**', '**=**' e '**R**' sono fissi, *n* è il numero del tasto numerico (1..9, 0) e *cc..cc* è il codice 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. [63](#)). 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* remote.

Nota: nel caso in cui siano abilitate le modalità "digitazione manuale del codice tessera" oppure "solo PIN" (vedi §10.2 a pag. [83](#)), non è più consentita la scelta rapida della causale tramite tasto numerico, pertanto il file FKEY.TXT viene usato solo per definire il menu di selezione "ridotto".

Esempio di FKEY.TXT che associa i due tasti numerici a sinistra (1 e 6) ai codici causali 11 e 22 e i due tasti a destra (5 e 0) ai codici causali 33 e 44, rispettivamente:

[Functions]

F1=R11

F6=R22

F5=R33

F0=R44

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 §4.10 a pag. [23](#)) è stato impostato al valore 3 (default 4). Può essere caricato solo via FTP, non si possono fare impostazioni via web server HTTP.

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.

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) collegato, rispettivamente, al connettore molex primario contrassegnato come "READER 1" (vedi figura della scheda a pag. 8), al connettore molex secondario contrassegnato come "FINGER BOX" (o "READER 2" nelle versioni hardware fino alla 005, vedi §3.8 a pag. 12) o alla morsettiera a vite estraibile contrassegnata come "EXTERNAL READER".

Si veda il "RFID2 13.56MHz Reader Manual" per una lista dei comandi disponibili per il modulo Mifare R&W.

Per quanto riguarda invece il modulo Legic R&W, l'unico comando disponibile è quello per definire le impostazioni di "autoread" del lettore, 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 0 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), **B:** BCD (Advant 0x01), **D:** 4 Bytes to 10 digits (Advant 0x05), **N:** Nessuno

Nota: è anche possibile inviare più comandi CR A... in sequenza, uno per ogni linea del file.

Se il lettore è stato impostato per comunicazione seriale TTL con configurazione di "autoread" custom (parametro **CardDecode=30 / 32 / 33 / 37** nella corrispondente sezione [Reader1], [Reader2] o [ExtReader] del file

PARAMETERS.TXT, vedi a pag. 33), 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. 40) 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** 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".

4.8 CONTROLLO REMOTO DEI RELE' E DEGLI INGRESSI DIGITALI DA WEB SERVER http

Mediante la sezione **"Remote Relays"** del web server HTTP è anche possibile pilotare remotamente il relé interno di ZP1/ZP2 (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. 29) 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 ZP1/ZP2 (quelli indicati con i numeri 1 e 2) 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); 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 del varco (descritti al §6.3 a pag. 69) hanno il valore "0" tranne **GateSensor1** che vale appunto "1", e tutti quelli che assegnano le uscite relé (descritti al §6.4 a pag. 71) hanno il valore "0" tranne **EntryRelay** e **ExitRelay** che valgono appunto "1".

X1/X2 Configuration

- Network
- GPRS modem
- FTP Client
- Time & Attendance
- Access Control
- Reader 1
- Reader 2
- External Reader
- Daylight Saving Time
- Time and Date
- USB
- System
- Remote Relays
- Biometrics
- File Manager
- Password
- Log

I/O's Control

Input status

Local		Remote			
		1		2	
Gate State 1	Not Assigned	Not Assigned	Not Assigned	Not Assigned	Not Assigned
0	0	-	-	-	-

Relays status

Local		Remote			
		1		2	
Entry Relay	Not Assigned	Not Assigned	Not Assigned	Not Assigned	Not Assigned
Exit Relay	Not Assigned	Not Assigned	Not Assigned	Not Assigned	Not Assigned
0	-	-	-	-	-

Select the device

Local
Local

☐ Open
☒ Close

20

* 100ms - Timeout ([2..255] 255 close permanently)

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. 67), 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"**.

X1/X2 Configuration

Network
GPRS modem
FTP Client
Time & Attendance
Access Control
Reader 1
Reader 2
External Reader
Daylight Saving Time
Time and Date
USB
System
Remote Relays
Biometrics
File Manager
Password
Log

I/O's Control

Input status

Local		Remote			
		1		2	
Gate State 1	Gate State 2	Gate Locked	Emergency	Not Assigned	Not Assigned
0	0	0	0	-	-

Relays status

Local		Remote		
		1		2
Entry Relay	Exit Relay	Gate Busy	Not Assigned	Not Assigned
0	0	0	-	-

Select the device

Select the output ☐ 1 ☒ 2

☐ Open

☒ Close * 100ms - Timeout ([2..255] 255 close permanently)

Dopo avere selezionato un dispositivo compaiono i *radio button* mediante i quali è possibile attivare o disattivare il relé corrispondente. **Nota:** Le voci "Close" e "Open" si riferiscono all'uscita normalmente aperta (**NO**), ma sia su ZP1/ZP2 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).

4.9 IMPOSTAZIONE PARAMETRI

Metodo con file di testo

Al primo riavvio del terminale dopo la formattazione della micro-SD (cosa che può essere fatta mediante l'opzione "Format SD card" nella pagina "System" del web server http di ZP1/ZP2), un file di testo ASCII chiamato **PARAMETERS.TXT** viene creato automaticamente, con tutti i valori di default dei parametri. Questo file viene anche ricreato automaticamente (sempre con tutti i valori di default) 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]*

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), ZP1 e ZP2 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. [114](#) per approfondire il concetto di messaggio "Keep Alive", ed il §12.4 a pag. [114](#) per sapere quali devono essere il formato della risposta del client HTTP ed i comandi di consultazione / impostazione del valore dei parametri.

4.10 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. 18), 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" monotoni 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" monotoni 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. 92) 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. 92), 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.

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. [92](#)) 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**

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** (nessuna registrazione in locale)

3 (default) → modalità semi-online: le transazioni vengono inviate online in HTTP al **MasterUrl**. 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. [118](#))

int **BufferSize**

Dimensione massima (in bytes) del buffer circolare per la revisione dati locale.

Valore di default: 500.000, Valore Max impostabile da menu HTTP: 99.999.999

int **RepeatTimeOut**

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

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

Max 99 (secondi)

int **DeleteOld**

0 (default): quando il file TRANSACTIONS.TXT ha superato le dimensioni specificate dal parametro **BufferSize** (vedi sopra), le successive transazioni non vengono più accettate, e viene mostrato il messaggio "Err. Memoria piena"

1: quando il file TRANSACTIONS.TXT ha superato le dimensioni specificate dal parametro **BufferSize** (vedi sopra), viene rinominato in "TRANSACTIONS.0.TXT". Ogni volta che supera nuovamente la dimensione massima, ciascun precedente file "TRANSACTIONS.n.TXT", se presente, viene rinominato in "TRANSACTION.(n+1).TXT", e il file corrente viene sempre rinominato in "TRANSACTIONS.0.TXT". Il più vecchio file di transazioni può essere "TRANSACTIONS.3.TXT", quindi in caso di successivi riempimenti il precedente file "TRANSACTIONS.3.TXT" viene automaticamente cancellato.

string **CustomRecord**

Definisce un formato personalizzato per le transazioni memorizzate nel file TRANSACTIONS.TXT (vedi §7.1 a pag. [77](#)).

Default: vuoto (viene utilizzato il formato standard, vedi §7 a pag. [75](#)). Lunghezza max: 68 caratteri.

string **Custom Entry**

Stringa che viene inserita nel campo “direzione di passaggio” (vedi §7.1 a pag. 77) 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. 52, e sono stati caricati i file CARDS.TXT, §5.4 a pag. 55, e USERS.TXT, §5.9 a pag. 61). 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*:

%hXX 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. 24, 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. 86), 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. 52), e sono stati caricati i file CARDS.TXT (§5.4 a pag. 55) e USERS.TXT (§5.9 a pag. 61), 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, ZP1/ZP2 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). 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 decodifiche alternative.

SEZIONE [AccessControl]

int **Enabled**

Abilita la funzionalità di controllo degli accessi (vedi §5 a pag. [52](#))

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

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 del varco è attivata (vedi parametro **GateEnabled** in questa stessa sezione a pag. [29](#)): 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 ZP1/ZP2 (default)

2, 3 → relé esterni, disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.3 a pag. [9](#))

int **ExitRelay**

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

int **PinOnly**

Abilita la modalità “solo PIN” (solo sui modelli X2 con tastiera numerica, vedi §5.12 a pag. [65](#))

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. [61](#))

0 → richiesta PIN disabilitata (default)

1 → richiesta PIN abilitata

Int **FullTable**

Usato solo quando ZP1/ZP2 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 (inclusi quelli risultati non validi secondo i criteri di controllo accessi). Le transazioni non valide 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. 75).

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

0 → vengono registrate solo le transazioni valide (default)

1 → vengono registrati tutti i tentativi di accesso

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 di un varco (vedi §6 a pag. 67)

0 → gestione varco non abilitata (default)

1 → abilita la gestione del varco

int **GateType**

Definisce il tipo di varco da gestire:

0 → varco non controllato (default). Se la gestione del varco è attivata (par. **GateEnabled**=1) ma il varco non è controllato, l'unica differenza rispetto al caso di gestione varco non attivata è che il messaggio "Keep Alive" inviato periodicamente all'host (vedi §6.5 a pag. 73) contiene sempre il server tag "**gateStatus=H**" (cioè "stato normale")

1 → porta battente: in questo caso l'input IN1 (già disponibile su ZP1/ZP2) è sempre utilizzato per lo stato della porta

2 → tornello: in questo caso l'input IN1 (già disponibile su ZP1/ZP2) è sempre utilizzato per lo stato del tornello

3 → doppia porta o bussola di sicurezza: in questo caso l'input IN1 (già disponibile su ZP1/ZP2) è sempre utilizzato per lo stato della prima porta, e l'input IN2 (anch'esso già disponibile su ZP1/ZP2) è sempre utilizzato per lo stato della seconda porta

Vedi anche il parametro **GateState1** a pag. 31 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. [61](#)).
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. [61](#)).
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)

2 → input IN2, già disponibile su ZP1/ZP2 (ad esclusione del varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.4 a pag. [10](#))

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)

2 → input IN2, già disponibile su ZP1/ZP2 (ad esclusione del varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.4 a pag. [10](#))

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)

2 → input IN2, già disponibile su ZP1/ZP2 (ad esclusione del varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.4 a pag. [10](#))

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 ZP1/ZP2

2, 3 → relé esterni, disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.3 a pag. [9](#))

int GateLocked

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

0 → non gestito (default)

2 → input IN2, già disponibile su ZP1/ZP2 (ad esclusione del varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.4 a pag. [10](#))

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 ZP1/ZP2

2, 3 → relé esterni, disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.3 a pag. [9](#))

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. [29](#), 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. [32](#)). I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → relé interno già disponibile su ZP1/ZP2

2, 3 → relé esterni, disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.3 a pag. [9](#))

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. [30](#)). I valori ammessi sono i seguenti:

0 → non gestito (default)

1 → relé interno già disponibile su ZP1/ZP2

2, 3 → relé esterni, disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.3 a pag. [9](#))

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)

2 → input IN2, già disponibile su ZP1/ZP2 (ad esclusione del varco di tipo doppia porta / bussola di sicurezza)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.4 a pag. [10](#))

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 ZP1/ZP2

2, 3 → relé esterni, disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.3 a pag. [9](#))

int **GateState1**

Definisce lo stato a riposo dell'input IN1, sempre usato per segnalare l'apertura / chiusura della porta o del tornello in caso di varco controllato (parametro **GateType** diverso da 0, vedi a pag. [29](#)).

0 → IN1 aperto (non attivo) con varco chiuso (default)

1 → IN1 cortocircuitato (attivo) con varco chiuso

int **GateState2**

Definisce lo stato a riposo dell'input IN2, sempre usato per segnalare l'apertura / chiusura della seconda porta nel caso di varco del tipo doppia porta o bussola di sicurezza (parametro **GateType**=3, vedi a pag. [29](#)).

0 → IN2 aperto (non attivo) con varco chiuso (default)

1 → IN2 cortocircuitato (attivo) con varco chiuso

int **ExternalNoTransit**

Ha effetto solo se è stato selezionato un varco del tipo tornello (parametro **GateType**=2, vedi a pag. [29](#)): 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)

2 → input IN2, già disponibile su ZP1/ZP2

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.4 a pag. [10](#))

int **TurnstileAlert**

Ha effetto solo se è stato selezionato un varco del tipo tornello (parametro **GateType**=2, vedi a pag. [29](#)): 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)

2 → input IN2, già disponibile su ZP1/ZP2

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.4 a pag. [10](#))

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. [29](#)): 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)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.4 a pag. [10](#))

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. [29](#)): 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)

3, 4 → input disponibili solo su scheda 914 NeoMAX opzionale (vedi §3.4 a pag. [10](#))

SEZIONE [Reader1]

int **CardDecode**

Selezione decodifica della carta:

0 (default): 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 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 (3+5=8 cifre decimali)

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

15: HID Clock&Data 37bit H10304 (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 (12bit facility + 18bit 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)

30: lettore RFID2 seriale TTL 13,56MHz – default: il formato e la lunghezza del codice letto dipendono dal tipo di carta e dalla configurazione di "autoread" del lettore, 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 seriale TTL HID iClass – 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 seriale TTL 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. 38), 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:** per un corretto funzionamento, il successivo parametro **BaudrateReader** deve essere impostato al valore di default (57600).
- 33: lettore seriale TTL 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 seriale TTL Legic – solo lettura del codice UID a 10 o 20 cifre decimali. **Attenzione:** ogni eventuale configurazione di “autoread” preimpostata diversa da questa verrà automaticamente rimossa; per un corretto funzionamento, si consiglia di impostare il successivo parametro **BaudrateReader** al valore 0 (autorilevamento della baudrate).
- 37: lettore RFID2 seriale TTL Legic con “autoread” custom: il formato e la lunghezza del codice letto dipendono dal tipo di carta e dalla configurazione di “autoread” del lettore, 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 seriale TTL Legic Tag A – valore custom riservato. **Attenzione:** ogni eventuale configurazione di “autoread” preimpostata diversa da questa verrà automaticamente rimossa.
- 51: Lettore generico con uscita Wiegand 26bit H10301 (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 (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 (12bit facility + 18bit 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)
- 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)

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 **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 14 cifre.

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

$n = 1, 2, \dots$ → 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

$n = 1, 2, \dots$ → 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 → codice comune non usato, 1 → prima posizione, etc.

int **FacilityCode**

Codice comune. Default : vuoto.

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]

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. [110](#) per ulteriori dettagli.

- 0 → verifica biometrica richiesta (salvo esenzioni per singoli codici tessera - default)
- 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. [28](#)) 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. [61](#))

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**

int **BaudrateReader**

int **CardCodeBegin**

int **CardCodeLength**

int **ShowCardCodeBegin**

int **ShowCardCodeLength**

int **FacilityCodeBegin**

int **FacilityCode**

int **Direction**

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**

int **BaudrateReader**

int **CardCodeBegin**

int **CardCodeLength**

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 **SkipBioVerify**

int **DisableFunctions**

int **ReaderLeds**

Presente solo nella sezione [ExtReader]. Consente di definire lo stato a riposo del LED rosso del lettore esterno (Nota: funziona solo se **WiegandOutput**=0 in questa stessa sezione).

0 → LED rosso spento a riposo

1 (default) → LED rosso acceso a riposo

SEZIONE [Biometric]

int **Enabled**

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

0 → gestione FingerBOX non attivata (default)

1 → attiva la gestione del modulo FingerBOX

int **FreeScan**

Attiva la modalità "autoscan" (o "identificazione 1:N", o "solo dito", vedi §11 a pag. [95](#) per ulteriori dettagli)

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

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. [95](#) 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 del sensore (utilizzo all'aperto o al chiuso). Può assumere solo i valori 0 (utilizzo all'aperto - default) o 1 (utilizzo al chiuso), e può avere effetto solo nel caso di sensore di impronte di tipo ottico, in quanto la luminosità dell'ambiente circostante (incidenza dei raggi luminosi, passaggi da luce a ombra) può influire sulle prestazioni del sensore, generando ad esempio delle scansioni "fantasma" in assenza del dito, o non rilevandone la presenza.

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. 96 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. 107). 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. 33).

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. 107), 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 §4.10 a pag. 24) 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. 110.

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

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. 55.

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

0 → 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 la 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.

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 → Spegnimento della retroilluminazione abilitato (default)

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

int **Contrast**

Contrasto del display (0..9) - Default 2

int **LogLevel**

Determina quanti e quali eventi, a seconda della loro importanza, vengono registrati nel file LOG.TXT (Nota: ogni volta che questo file eccede la dimensione massima di 512KB, viene automaticamente rinominato in LOG.0.TXT, e gli eventuali file precedenti vengono a loro volta rinominati, fino al file più vecchio mantenuto in memoria che è sempre LOG.3.TXT).

0: dettaglio massimo (tutti gli eventi vengono registrati)

1: dettagliato

2: solo gli eventi principali vengono registrati (default)

3: file LOG.TXT non utilizzato

int **TTY1Config**

Flag 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.

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**

Determina il set di caratteri (e quindi la relativa tabella di codifica Windows-125x, vedi §18 a pag. [129](#)) da utilizzare per la visualizzazione dei messaggi a display:

0: Europa occidentale (Windows-1252) - default

1: Turco (Windows-1254)

2: Europa centro-orientale (Windows-1250)

string **OperatorPassword**

Password per accedere al menu supervisore dalla tastiera del terminale - Default 00000

string **RemotePassword**

Password per l'accesso remoto da un client FTP o browser web – Default "admin"

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. [80](#)) 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: volume alto (default)

2: volume medio

3: volume basso

int **TTY1Legacy**

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

string **FirmwareKey**

Chiave di attivazione per funzioni opzionali del firmware. Vedi §4.11 a pag. 50 per una descrizione dettagliata. Default: vuoto.

string **Firmware**

Stringa identificativa della versione corrente del firmware nel formato "aNNbuildnnnn".

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). **Nota:** questi valori vengono controllati solo alla mezzanotte di ogni giorno, pertanto il cambio di orario avviene correttamente solo se ZP1/ZP2 è operativo alle 00:00 del giorno in cui dovrà essere effettuato il cambio.*

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**

1 → Imposta automaticamente i passaggi all'ora legale/solare (default): alle 02:00 dell'ultima domenica di marzo l'orologio viene portato avanti di un ora per il passaggio all'ora legale (DST), mentre alle 03:00 dell'ultima domenica di ottobre l'orologio viene portato indietro di un ora per il ritorno all'ora solare (funzionamento valido per i paesi che adottano l'orario CET=UTC/GMT+1)

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

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 → campo DAYLIGHT vuoto (default)

string **UTC**

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

Default: +00:00

int **RecordUTC**

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

0 → il campo UTC è vuoto (default)

SEZIONE [Ethernet]

int **DHCP**

1: DHCP ON (default)

0: DHCP OFF

string **IPAddress**

Valore di default 192.168.1.240 (usato solo se il DHCP è OFF, o se il server DHCP non risponde)

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 con protocollo FTP. Default: 21.

string **MasterUrl**

Indirizzo IP o URL logico dell'host, default 0.0.0.0 (porta 80 per il protocollo HTTP, 8499 per il "keep alive" UDP). 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 ZP1/ZP2 venga gestito dal *middleware* XAM o 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. [111](#). Il protocollo FTP è comunque sempre attivo.

string **httpOnlineMessage**

Vedi §12 a pag. [111](#) per una descrizione dettagliata

string **httpBatchMessage**

Vedi §12 a pag. [111](#) per una descrizione dettagliata

string **httpKeepAliveMessage**

Vedi §12 a pag. [111](#) 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. [4](#))

int **ConnTimeout**

Timeout in collegamento ONLINE, default 5 secondi

int **KeepAliveInterval**

Vedi §12 a pag. [111](#) 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: Server HTTP abilitato (default)
0: Server HTTP disabilitato

int **EnableFTPServer**

1: Server FTP abilitato (default)
0: Server FTP disabilitato

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

SEZIONE [GPRS]

int **Enabled**

Attiva la gestione del modem GPRS integrato opzionale.
0: Modem GPRS disabilitato (default)
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 utilizzo del protocollo FTP)
9999: il modem effettuerà la connessione GPRS solo se vi sono esportazioni schedate tramite il client FTP del terminale (vedi §7.3 a pag. [79](#)), e solo in corrispondenza degli orari impostati, disconnettendosi automaticamente al termine di ciascuna sessione FTP.
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. [124](#)).

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 seguente:

991#**

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 [FTPCCLIENT]

string **ServerUrl**

Contiene l'indirizzo del server FTP a cui inviare il file TRANSACTIONS.TXT corrente (vedi §7.3 a pag. [79](#)) agli orari schedulati tramite il file ALARMS.TXT (vedi §4.2 a pag. [17](#)). Può essere sia un indirizzo IP che un URL logico. In caso il server sia configurato su una porta differente dal default (porta 21), è possibile specificarla dopo l'indirizzo del server stesso. 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: Modalità passiva abilitata (default)

int RetryNumber

Indica il numero di tentativi 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

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 (solo su versioni di hardware 006 e successive, vedi §14 a pag. [120](#)).

0: Gestione USB disabilitata (default)

1: Gestione USB abilitata

string PasswordUSB

Password per accedere al menu di gestione delle chiavette di memoria USB in seguito all'inserimento della chiavetta - Default 00000

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. [25](#) 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)

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
BufferSize=500000
RepeatTimeOut=0
DeleteOld=0
CustomRecord=""
CustomEntry="1"
CustomExit="0"
BeepOnCard=0
ScreenOk=""
ScreenError=""
HideTypedCode=0
MultiFormat=0
```

```
[AccessControl]
Enabled=0
RelayActivation=5
EntryRelay=1
ExitRelay=1
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

[Reader1]
CardDecode=0
BaudrateReader=57600
CardCodeBegin=0
CardCodeLength=6
ShowCardCodeBegin=0
ShowCardCodeLength=0
FacilityCodeBegin=0
FacilityCode=
Direction=2
SkipBioVerify=0
DisableFunctions=0

[Reader2]
CardDecode=0
BaudrateReader=57600
CardCodeBegin=0
CardCodeLength=6
ShowCardCodeBegin=0
ShowCardCodeLength=0
FacilityCodeBegin=0
FacilityCode=
Direction=2
DisableFunctions=0

[ExtReader]
CardDecode=0
BaudrateReader=57600
CardCodeBegin=0
CardCodeLength=6
ShowCardCodeBegin=0
ShowCardCodeLength=0

FacilityCodeBegin=0
FacilityCode=
WiegandOutput=0
Direction=2
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

[System]
TurnOffTimeout=10
TurnoffBacklight=1
Contrast=2
LogLevel=2
TTY1Config=1
Language=English
FontEncoding=0
OperatorPassword=00000
RemotePassword=admin
TimeLock=
AudioVolume=1
TTY1Legacy=1
FirmwareKey=
Firmware=aNNbuildnnnn

[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

[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

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=\$localtransactio
n\$

TermID="X1"

ConnTimeout=5

KeepAliveInterval=15

RetryTimeout=60

EnableHTTPServer=1

EnableFTPServer=1

[GPRS]

Enabled=0

ConnectionInterval=0

ATextraCommand=""

Dialnum=""

User=""

Password=""

[FtpClient]

ServerURL=

User=""

Password=""

PassiveMode=1

RetryNumber=3

DestinationFileName="TRANSACTIONS.TXT"

FileOpeningMode=0

[USB]

Enabled=0

PasswordUSB=00000

TrnsFileUSB=TRANSACTIONS.TXT

MoveTrnsToUSB=1

4.11 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. 88), 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):

```
Ethernet
00:04:24:A1:DC:2B [00:A3]
DHCP: On
IP: 192.168.1.122
SM: 255.255.255.0
GW: 192.168.1.254
```

Le stesse informazioni appaiono anche temporaneamente (per circa 3 secondi) nella seconda schermata mostrata all'avvio del terminale (vedi §10.1 a pag. 82).

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

X1/X2 Configuration

Network	System
GPRS modem	Firmware X1 a08 build 222, Jun 28 2012 12:23:06
FTP Client	Bootloader 1.4
Time & Attendance	MAC Address 00:04:24:A3:5C:2B [CA,F1]
Access Control	Available Free Space 950 MBytes
Reader 1	Battery 5685 mV - Normal
Reader 2	Server Offline - Pending Record 0
External Reader	Screen Snapshot <input type="button" value="Snapshot"/>
Daylight Saving Time	Restart Terminal <input type="button" value="Restart"/>
Time and Date	Format SD Card <input type="button" value="Format"/>
USB	Reset default parameters <input type="button" value="Reset"/>
System	Recover all the transactions <input type="button" value="Recover"/>
Remote Relays	Language <input type="text" value="English"/>
Biometrics	Font encoding <input type="text" value="Western European - Windows-1252"/>
File Manager	Audio volume <input type="text" value="High"/>
Password	Timeout on Battery <input type="text" value="10"/> minutes
Log	Turn Off Backlight on Battery <input checked="" type="checkbox"/>
	TTY1 Legacy <input checked="" type="checkbox"/>
	Display Contrast <input type="text" value="2"/>
	Log Level <input type="text" value="2"/> 0=Verbose, 3=Not Active
	Operator Password <input type="text" value="0000"/>
	Time Lock <input type="text" value="0000"/> - <input type="text" value="00"/> - <input type="text" value="00"/> YYYY-MM-DD
	Firmware Key <input type="text"/>
	<input type="button" value="Save"/>

- 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:

00:04:24:A1:DC:2B:DD:A3

Lista dei moduli firmware opzionali da attivare

Al momento attuale l'unico modulo firmware opzionale esistente è quello relativo alla gestione di un varco di controllo accessi, la cui denominazione esatta è "ZP1/ZP2 GATE MANAGER". **Nota:** se ZP1/ZP2 viene gestito dal programma XatI@s, la chiave di attivazione per la gestione del varco viene caricata automaticamente, quindi non è necessario farne richiesta né inserirla manualmente.

Inserimento della chiave di attivazione

Ogni chiave di attivazione è una stringa di 16 cifre esadecimali (ad es. 20EB0238FFFFFFF), 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 vengano effettuate le operazioni di formattazione della SD card o il reset dei parametri 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, oppure un *downgrade* del firmware ad una versione "a07_build641" o precedente, questa impostazione viene rimossa.

5. TABELLE DI CONTROLLO ACCESSI

Impostando a 1 il parametro **Enable** nella sezione [AccessControl] del file PARAMETERS.TXT (vedi §4.10 a pag. 28) 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 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 ZP1/ZP2 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.

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. 53). 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 ZP1 e ZP2 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. 52). 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. 65).

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. 17 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. 52) lo stesso significato che il file **HOLIDAYS.TXT** (§4.3 a pag. 17) ha per **ALARMS.TXT** (§4.2 a pag. 17), 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. 65), 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. 52. 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 67 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. 58). 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. 61). 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. 54):

R_CCCCCCCCCCCCCC_GGGGGGGGGG_IIIIIIII_AAMMGHHMM_aammgghmm_E_00**CR LF**

oppure

R_CCCCCCCCCCCCCC_GGGGGGGGGG_IIIIIIII_AAMMGHHMM_aammgghmm_E_00_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. 94); se si utilizza la versione con flag biometrico, invece, e se il parametro **EnrollAuth=1** nella sezione *[Biometric]* del file PARAMETERS.TXT (vedi §4.10 a pag. 39), 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.
 - 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. [102](#)).

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. [58](#)). 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 influente).

IIIIIIII

10 byte che rappresentano l’identificatore univoco dell’utente associato alla tessera (definito nel file USERS.TXT, vedi §5.9 a pag. [61](#)). 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.

AAMMGHHMM

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.

aammgghmm

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).
00	2 byte che rappresentano il codice di "edizione" della tessera (<u>campo attualmente non gestito</u> : lasciare fisso a '00').
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. 94). '1' significa tessera autorizzata all'utilizzo della biometria, '0' significa tessera non autorizzata. Nota: ha effetto solo se il parametro EnrollAuth =1 nella sezione <i>[Biometric]</i> del file PARAMETERS.TXT (vedi §4.10 a pag. 39).

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, ed è abilitata all'accesso. In questo caso il flag biometrico non è stato usato.

5.5 CARDNGE.TXT

Contiene l'elenco degli intervalli di codici tessera abilitati all'accesso. Ogni record ha una lunghezza fissa di 81 byte (cioè 79 caratteri + CRLF) 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. 58), secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. 54):

RRRRRRRRRRR_cccccccccccccc_CCCCCCCCCCCCCC_GGGGGGGGGG_AAMMGHMM_aammghmm_ECRLF

Dove:

RRRRRRRRRRR	10 byte che rappresentano l'identificatore univoco dell'intervallo di codici tessera (<u>chiave primaria</u>). Questo dato è richiesto per ogni record e deve essere univoco all'interno del file e diverso da "000...000".
cccccccccccccc	16 byte che rappresentano il limite <u>inferiore</u> 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.
CCCCCCCCCCCCCC	16 byte che rappresentano il limite <u>superiore</u> 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. 58). 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 influente).

AAMMGHMM

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. 55) 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.

aammgghmm

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. 55) 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. 59), secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. 54):

```
GGGGGGGGGG_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_
IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_
IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_
IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_
IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_
IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_IIIIIIII_
```

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. 55) e/o CARDRNGE.TXT (vedi §5.5 a pag. 57). Questo dato è richiesto per ogni record e deve essere univoco all'interno del file e diverso da "000...000".

11/11/2019

10 byte che rappresentano l'identificatore univoco dell'autorizzazione (definito nel file AUTH.TXT (vedi §5.7 a pag. [59](#)). 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:

[illegible]

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 ZP1 e ZP2 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. [60](#)), secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. [54](#)):

```
IIIIIIII_0000000001_MMMMMMMMMM_MMMMMMMMMMM_MMMMMMMMMM_MMMMMMMMMM_MM  
MMMMMMMMM_  
MMMMMMMMMMM MMMMMMMMMMM MMMMMMMMMMM CR LF
```

Dove:

11/11/2019

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. [58](#)). Questo dato è richiesto per ogni record e deve essere univoco all'interno del file e diverso da "000...000".

000000001

Stringa fissa di 10 byte che rappresenta l'identificatore dell'unico terminale gestito dall'applicazione di controllo accessi.

MMMMMMMMMMMM

10 byte che rappresentano l'identificatore univoco del modello orario di riferimento (definito nel file TIMEMOD.TXT, vedi §5.8 a pag. [60](#)). 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_0000000000
000 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. 54):

```
MMMMMMMMMMMM_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 CR LF
```

oppure

```

MMMMMMMMMMMMMMMM_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. [59](#)). 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 così 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. 65).

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:

[illegible]

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. [63](#)) e ad un codice PIN, secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. [54](#)):

IIIIIIIII_PPPP_NNNNNNNNNNNNNNNNNNNNNN_AAMMGGHHMM_aammgghhmm_E_T_UUUUUUUUUU CR LF

Dove:

11/11/2023

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. 55). 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 §4.10 a pag. 28) 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 §4.10 a pag. 36). Inoltre, se è abilitata la modalità "solo PIN" (solo sui modelli X2 con tastiera numerica, vedi §5.12 a pag. 65), 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:** 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.

NNNNNNNNNNNNNNNNNNNNNN

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-125x mostrate al §18 a pag. 129.

AAMMGHHMM

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. 55) 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. 55) 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 del varco sia attivata, vedi par. §6 a pag. 67). '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 §4.10 a pag. 29); '1' significa utente a cui sono assegnati i tempi estesi, definiti dagli analoghi parametri **TimeOutOpenExtended** e **TimeOutCloseExtended**.

UUUUUUUUUUUUUUUU

10 byte che rappresentano la tipologia dell'utente. Questo codice può essere usato nel file AXREASON.TXT (vedi §5.10 a pag. [63](#)) 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.

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. [61](#)) secondo il seguente formato (tenete sempre presente le regole di validità generale descritte al §5.3 a pag. [54](#)):

```
IIIIIIIII_CCCCCCCCCC_00_DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD_E_UUUUUUUUUUUU_UUUUUUUUUUUU_
UUUUUUUUUUUU UUUUUUUUUUU FCR LF
```

Dove:

11/11/2019

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".

CCCCCCCCC

10 byte che rappresentano il codice causale. 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. [75](#)). 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').

DD

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. 92). 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-125x mostrate al §18 a pag. 129.

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. 61). 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. 55), e associati ad utenti definiti nel file USERS.TXT (vedi §5.9 a pag. 61) 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. 17) 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).

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. [60](#)). 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. [54](#)):

GGMAAA GGMAAA GGMAAA ...GGMAAA CR 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. [60](#)). 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:

[illegible]

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 §4.10 a pag. [28](#)) 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. 55) e USERS.TXT (vedi §5.9 a pag. 61) 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 §4.10 a pag. 28), 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. 83 (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	File CARDS.TXT e CARDRNGE.TXT assenti <i>oppure</i> Codice tessera non presente in CARDS.TXT e non compreso in un intervallo di CARDRNGE.TXT <i>oppure</i> 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 <i>oppure</i> Codice tessera compreso in un intervallo associato ad un periodo temporale scaduto o non ancora iniziato in CARDRNGE.TXT <i>oppure</i> 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) <i>oppure</i> 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) <i>oppure</i> 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 <i>oppure</i> 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
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) <i>oppure</i> 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

6. GESTIONE DI UN VARCO

Mediante l'inserimento di un'apposita chiave di attivazione FW "ZP1/ZP2 GATE MANAGER" (vedi §4.11 a pag. 50), è possibile sbloccare la funzionalità di gestione di un varco di controllo accessi. Solo nel caso in cui ZP1/ZP2 venga gestito dal programma Xatl@s, non è necessario richiedere la chiave né inserirla manualmente, poiché essa viene caricata in maniera automatica.

Una volta inserita la relativa chiave di attivazione, impostando a 1 il parametro **GateEnabled** nella sezione [AccessControl] del file PARAMETERS.TXT (vedi §4.10 a pag. 29) oppure, analogamente, spuntando la checkbox "**Gate Enabled**" nella pagina "**Access Control**" del web server HTTP, si attiva la funzionalità di gestione del 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. 73), 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 ZP1/ZP2 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).

I parametri per la gestione del varco si trovano anch'essi nella sezione [AccessControl] del file PARAMETERS.TXT (vedi §4.10 a pag. 29) 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. 73.

6.1 TIPO DI VARCO

- **GateType**

Definisce il tipo di varco da gestire:

- 0 → varco non controllato (default). Se la gestione del varco è attivata (par. **GateEnabled**=1) ma il varco non è controllato, l'unica differenza rispetto al caso di gestione varco non attivata è che il messaggio "Keep Alive" inviato periodicamente all'host (vedi §6.5 a pag. 73) 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. 69, 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. 69, 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. 69, l'input IN1 viene automaticamente assegnato allo stato della prima porta, e l'input IN2 allo stato della seconda porta

Come si può vedere, in tutti i casi di varco controllato, per default l'input IN1 (già disponibile su ZP1/ZP2) 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. 69. Solo nel caso del varco di tipo doppia porta o bussola di sicurezza, inoltre, per default anche l'IN2 (anch'esso già disponibile su ZP1/ZP2) 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. 68) o ad uno sblocco comandato (vedi §6.3 a pag. 69 e §6.5 a pag. 73), 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 ZP1/ZP2 mostra il messaggio "**Effrazione**" e registra nel file TRANSACTIONS.TXT un record relativo all'emissione di un evento, con il campo **EVENTO**="11" (vedi §7.2 a pag. 78). 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, ZP1/ZP2 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).

ZP1/ZP2 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 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. 75).
- **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. 61). Default: 300 (30 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: 100 (10 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. 75). Inoltre nello stesso file viene registrato anche un record relativo all'emissione di un evento, con il campo **EVENO="12"** (vedi §7.2 a pag. 78). 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. 61). Default: 300 (30 secondi).

6.3 ASSEGNAZIONE DEGLI INGRESSI DIGITALI

Come visto al §6.1 a pag. 67, in tutti i casi di varco controllato, per default l'input IN1 (già disponibile su ZP1/ZP2) viene utilizzato per segnalare lo stato del varco (aperto / chiuso), ma è comunque possibile assegnarlo ad un controllo di tipo diverso mediante uno degli altri 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 ZP1/ZP2) viene usato per lo stesso scopo ma relativamente alla seconda porta, e comunque anch'esso potrà essere riassegnato.

Su un terminale ZP1/ZP2 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 ZP1/ZP2 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 ZP1/ZP2 (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 ZP1/ZP2 (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. 67).

- **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. 67).

- **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. 71, per il tempo definito dal parametro **RelayActivation**, vedi §4.10 a pag. 28). 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. 75).

- **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. 71, per il tempo definito dal parametro **RelayActivation**, vedi §4.10 a pag. 28). 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. 75).

- **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. 71). 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 §7.2 a pag. 78). 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 §7.2 a pag. 78), inoltre attiva il relé eventualmente definito dall'omologo parametro **GateLockedRelay**, vedi §6.4 a pag. 71). 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. 71. 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. 67): 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. 75).

- **TurnstileAlert**

Ha effetto solo se è stato selezionato un varco del tipo tornello (parametro **GateType**=2, vedi §6.1 a pag. 67): 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 §7.2 a pag. 78). 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. 67), pertanto i valori ammessi sono solo i seguenti:

0 → non gestito (default)

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'

- **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 RELÉ

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 ZP1/ZP2 (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 ZP1/ZP2 preso singolarmente, il quale dispone di 1 sola uscita relé, **EntryRelay** e **ExitRelay** devono necessariamente essere impostati entrambi a "1", e nessuno dei 5 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 ZP1/ZP2 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 5 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 5 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 5 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 5 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 ZP1/ZP2

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'

- **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. 69).

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. 69). 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 dell'input IN1 (sempre associato allo stato della porta o del tornello, vedi parametro **GateType** al §6.1 a pag. 67, 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 all'omologo parametro **TurnstileAlert** (effrazione, solo in caso di varco di tipo tornello, vedi §6.3 a pag. 67). 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. 67). 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. 67) 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. [111](#); ZP1/ZP2 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 del varco è attiva, ZP1/ZP2 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. [114](#), 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. [67](#)), 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. [68](#)) 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. [111](#)) 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. [75](#)). 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. [114](#)), 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. 69). 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. 75), **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. 69). 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. 69). 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. 69). 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 §7.2 a pag. 78).

7. TRANSAZIONI

Se il parametro **Offline** all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi §4.10 a pag. 25) 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. 113), 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. 118 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 §7.1 a pag. 77): 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 DELLA LIENA ZUCCHETTI 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 (1 GB 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,CODICE_CAUSALE,CODICE_UTENTE,CONTROLLI,ESITO,SOURCE,UTC,DAYLIGHT,...
(i campi in grassetto sono sempre presenti, 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

CODICE_CAUSALE, (default vuoto)

CODICE_UTENTE codice utente (numerico o alfanumerico) estratto dalla carta secondo i parametri **CardCodeBegin** e **CardCodeLength**, vedi §4.10 a pag. 35).

Nota: Questo campo viene riempito con soli zeri "000..000" in caso di gestione del varco attiva e sblocco da pulsante manuale o da comando online per singolo transito (vedi §6.3 a pag. 69 e §6.5 a pag. 73).

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. 52) 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 §4.10 a pag. 29) 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. 66:

00 = OK

33 = Tessera non valida
 34 = 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
 46 = (*)Tab Err (Tabella mancante)
 51= Verifica biometrica 1:1 fallita
 52= Tessera scaduta
 53= Mancano entrambe le tabelle CARDS.TXT e CARDNRNGE.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 del varco sono abilitati), altrimenti fisso a "0":

0 = Transazione avvenuta
 1 = Transazione non avvenuta, poiché non consentita (se il campo CONTROLLI è diverso da "00")
oppure,
 ma solo se è stata attivata la gestione di un varco (vedi §6 a pag. 67), per transito non effettuato (varco non aperto anche se sbloccato, in questo caso il campo CONTROLLI="00")

SOURCE

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 §4.10 a pag. 24), 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)

*Campi opzionali (vedi sezione **TimeSettings** al §4.10, pag. 41):*

UTC

Differenza fra il fuso orario locale e quello universale UTC/GMT
 +HHMM se la differenza è positiva
 -HHMM se la differenza è negativa
 Z se il terminale si trova nel fuso orario del meridiano di Greenwich

DAYLIGHT

0= transazione effettuata durante l'orario solare, 1= ora legale

RFU, ... altri campi che potranno essere aggiunti nei rilasci futuri

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 ZP1/ZP2.

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. 79.

Il file TRANSACTIONS.TXT può anche essere copiato manualmente su una chiavetta di memoria USB, seguendo le modalità illustrate al §14 pag. 120.

Tutte le transazioni registrate in locale possono comunque essere scaricate, record per record, mediante un client HTTP in modalità *batch* (vedi §12 e §12.5 a pag. 118 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 scaricate 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 §4.10 a pag. 25) è impostato a 1, il terminale si comporta come segue: quando TRANSACTIONS.TXT ha superato la dimensione specificata dal parametro **BufferSize** (vedi §4.10 a pag. 25), o quando il file riservato **btransactions.loc** ha superato il doppio della dimensione specificata dal parametro **BufferSize**, TRANSACTIONS.TXT viene rinominato in “TRANSACTIONS.0.TXT”. Ogni volta che uno dei 2 file supera nuovamente la rispettiva dimensione massima, ciascun precedente file “TRANSACTIONS.n.TXT”, se presente, viene rinominato in “TRANSACTION.(n+1).TXT”, e il file corrente viene sempre rinominato in “TRANSACTIONS.0.TXT”. Il più vecchio file di transazioni può essere “TRANSACTIONS.3.TXT”, quindi in caso di successivi riempimenti il precedente file “TRANSACTIONS.3.TXT” viene automaticamente cancellato. Analogamente, quando TRANSACTIONS.TXT ha superato la dimensione specificata dal parametro **BufferSize**, il file riservato **btransactions.loc** viene rinominato in “btransaction.0.loc”, ma solo se non vi sono transazioni HTTP “pendenti”: in caso contrario non viene rinominato finché il file TRANSACTIONS.TXT non viene cancellato di proposito (in questo caso si assume che la ricezione di tutte le transazioni sia comunque avvenuta via FTP), oppure finché lo stesso **btransactions.loc** non ha superato il doppio della dimensione specificata dal parametro **BufferSize**. In entrambi i casi, quando questo file viene rinominato viene anche azzerato il contatore delle timbrature “pendenti”. Come per “TRANSACTIONS.3.TXT”, non ci può essere un file più vecchio di “btransactions.3.loc”.

Se invece il parametro **DeleteOld** è impostato a 0, il terminale si comporta come segue: quando il file TRANSACTIONS.TXT ha superato le dimensioni specificate dal parametro **BufferSize**, il terminale si rifiuta di registrare ogni nuova transazione mostrando il messaggio “Err. Memoria piena”. Se invece è il file riservato **btransactions.loc** a superare la rispettiva dimensione massima, non cambia nulla rispetto al caso **DeleteOld**=1.

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. 128.

7.1 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 della sezione [TimeAttendance] del file PARAMETERS.TXT (vedi §4.10 a pag. 25).

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 TRAXi32), 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 §4.10 a pag. 43)
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 mediante l'impostazione dei parametri CustomEntry e CustomExit all'interno della sezione <i>[TimeAttendance]</i> del file PARAMETERS.TXT (vedi §4.10 a pag. 25).
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. 17) sono più lunghi ne verrà registrata solo la parte meno significativa, 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 §4.10 a pag. 35) verrà registrata solo la parte meno significativa del codice utente letto, se invece è maggiore il campo verrà riempito con zeri a sinistra.
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 SOURCE descritto al §7, pag. 75)
\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>.

7.2 EMISSIONE E RIENTRO DI EVENTI RELATIVI ALLA GESTIONE DI UN VARCO

Se è stata attivata la gestione di un varco (vedi §6 a pag. [67](#)) e se ZP1/ZP2 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 ",":

AAAAAMMGG,HHMMSS,DIREZIONE,,,EVENTO,0,7,UTC,DAYLIGHT,....

(i campi in grassetto sono sempre presenti, quelli in corsivo sono opzionali e hanno lo stesso significato di quelli visti al §7)

AAAAAMMGG 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

DIREZIONE 5=emissione, 6=rientro

EVENTO tipologia dell'evento:

03 = Varco disabilitato (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. [69](#)

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. [69](#)

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

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. [73](#)) 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.

7.3 INVIO DELLE TRANSAZIONI TRAMITE CLIENT FTP

A partire dalla versione di firmware a08_build018, il terminale include un client FTP che permette di esportare automaticamente il file TRANSACTIONS.TXT corrente verso 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 §4.10 a pag. [44](#), oppure collegandosi tramite web browser al terminale, come indicato al §4 pag. [13](#).

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, le opzioni relative ai tentativi in caso di fallimento 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 §4.10 a pag. [44](#).



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 X1_FTP_TEST.TXT creato utilizzando le stesse modalità di invio che verranno utilizzate durante le esportazioni schedate.

La schedulazione delle esportazioni deve essere configurata manualmente editando il file ALARMS.TXT come indicato al §4.2 a pag. 17.

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 schedato 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 schedata, 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.

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, portoghese, 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 nuove 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 ZP1/ZP2, 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 determinata dal valore del parametro **Language** all’interno della sezione [System] del file PARAMETERS.TXT (vedi §4.10 a pag. 40), 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”.

X1/X2 Configuration

Network	System
Time & Attendance	Firmware X1 a07 build 783, Jan 23 2012 13:00:39
Access Control	Bootloader 1.4
Reader 1	MAC Address 00:04:24:A1:DC:2B [DD,A3]
Reader 2	
External Reader	Available Free Space 1908 MBytes
Daylight Saving Time	Battery 5786 mV - Normal
Time and Date	Server Offline - Pending Record 40
USB	Screen Snapshot <input type="button" value="Snapshot"/>
System	Restart Terminal <input type="button" value="Restart"/>
Remote Relays	Format SD Card <input type="button" value="Format"/>
Biometric	Reset default parameters <input type="button" value="Reset"/>
File Manager	Recover all the transactions <input type="button" value="Recover"/>
Password	
Log	

Language	English
Timeout on Battery	English
Turn Off Backlight on Battery	Italiano
Display Contrast	Português
Log Level	Español
Operator Password	00000
Time Lock	0000 - 00 - 00 YYYY-MM-DD
Firmware Key	20EB0238FFFFFFFF

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.

9. AGGIORNAMENTO FIRMWARE

Gli aggiornamenti firmware di ZP1/ZP2 sono disponibili come file chiamati "**XONE_ann_buildxxx.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_ann_buildxxx.bin**" contenente la nuova versione di firmware sul file system del terminale via FTP, come fareste con qualunque altro file.

Il terminale riconoscerà automaticamente la presenza di un nuovo firmware ed effettuerà le operazioni necessarie. 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.**

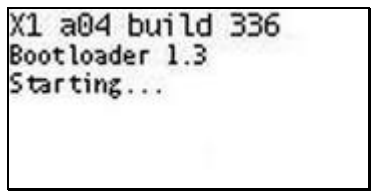
E' anche possibile aggiornare il firmware in locale mediante una chiavetta USB, come descritto al §14.4 a pag. [122](#) (utile per terminali *stand-alone* non collegati in Ethernet, o su cui comunque non sia possibile caricare il firmware via FTP).

10. INTERFACCIA UTENTE DI ZP1/ZP2

10.1 AVVIO

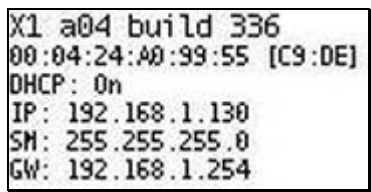
Un terminale correttamente alimentato si accende da solo. L'avvio del sistema richiede circa 6 secondi. Per riavviare ZP1/ZP2 tenere premuto il pulsante  per circa 6 secondi, o premere brevemente il pulsante RESET sulla scheda (vedi §3.2 a pag. 8). Il pulsante  permette anche di spegnere il terminale, ma solo se sta funzionando a batteria. Se si spegne durante il funzionamento a batteria, ZP1/ZP2 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:



```
X1 a04 build 336  
Bootloader 1.3  
Starting...
```

e poi la configurazione Ethernet, ancora per 3 secondi:



```
X1 a04 build 336  
00:04:24:A0:99:55 [C9:DE]  
DHCP: On  
IP: 192.168.1.130  
SN: 255.255.255.0  
GW: 192.168.1.254
```

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)

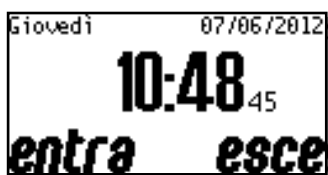
L'aspetto dello schermo può essere cambiato agendo sui seguenti parametri: **DirMode**, **CompanyName**, **SecondsShown**, **AmPm**, **MonthDay**, **DateSeparator**.



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 un 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. 92)

▼ → menu dei codici causale, solo se è stato precedentemente caricato un file di testo chiamato REASONS.TXT (vedi §4.4 alla pag. 17), o in alternativa un altro file chiamato AXREASON.TXT (vedi §5.10 alla pag. 63)

[<-]> → inversione direzione di transito (solo se il parametro DirMode all’interno della sezione [TimeAttendance] del file PARAMETERS.TXT, vedi §4.10 a pag. 23, 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 §4.10 a pag. 24) 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. 27) 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. 65). 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. 87 (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. 18) e almeno uno fra REASONS.TXT (vedi §4.4 alla pag. 17) e AXREASON.TXT (vedi §5.10 alla pag. 63), 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)

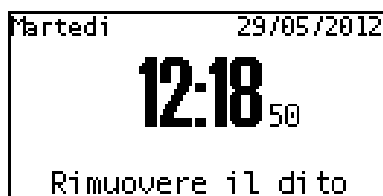
↩ → menu “ridotto” per la selezione delle sole causali e/o delle *enquiries* remote (queste ultime disponibili solo se ZP1/ZP2 viene gestito dal programma Xatl@s) associate a tasti numerici per la selezione diretta (scelta rapida), solo se è stato precedentemente caricato il file FKEY.TXT (vedi §4.5 alla pag. 18)

AUTENTICAZIONE BIOMETRICA

Se ZP1/ZP2 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. 95), 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 quelle già registrate nel modulo. Non appena il sensore rileva la presenza del dito, ZP1/ZP2 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 ZP1/ZP2 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. 38) 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 ZP1/ZP2 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 tessera utilizzato, non siano stati esentati dalla verifica biometrica, con le modalità descritte al §11.1 a pag. 100 o al §11.4 a pag. 110), confrontando l'impronta appena scansionata solo con quelle già registrate per il codice tessera introdotto:

Cod: 000043
Appoggia il dito

Solo se sono stati caricati i file CARDS.TXT (§5.4 a pag. [55](#)) e USERS.TXT (§5.9 a pag. [61](#)), invece del codice della tessera ZP1/ZP2 visualizza il nome dell'utente ad esso relativo:

Mario Rossi
Appoggia il dito

Se entro 10 secondi non viene rilevata la presenza del dito, ZP1/ZP2 abortisce la transazione e mostra il messaggio di errore **“Operazione annullata”**, altrimenti procede con la verifica di identità e mostra il seguente messaggio:

Cod: 000043
Rimuovere il dito

Entro un secondo, ZP1/ZP2 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. [55](#).

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**:

Domenica	20/03/2011
18:10 33	
Entrata: 000646	

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. [35](#)) 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. 35).

Solo se il controllo degli accessi è stato attivato (vedi §5 a pag. 52), e se sono stati caricati i file CARDS.TXT (§5.4 a pag. 55) e USERS.TXT (§5.9 a pag. 61), in caso di transazione accettata è possibile visualizzare, invece del codice della tessera, il nome dell'utente ad esso relativo:



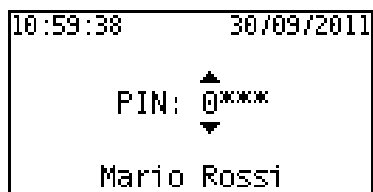
Nota: se il parametro **HideTypedCode** all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT (vedi pag. 27) è stato impostato a 1 (default 0) per mascherare la digitazione manuale del codice personale, quest'ultimo non viene neanche mostrato per conferma in caso di transazione accettata (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" monotoni 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. 24).

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. 26).

10.4 RICHIESTA CODICE PIN

Se il controllo degli accessi è stato attivato (vedi §5 a pag. 52) e sono stati caricati i file CARDS.TXT (§5.4 a pag. 55) e USERS.TXT (§5.9 a pag. 61), e se il parametro **AskPin** all'interno della sezione *[AccessControl]* del file PARAMETERS.TXT (vedi §4.10 a pag. 28) 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 §4.10 a pag. 36), 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.

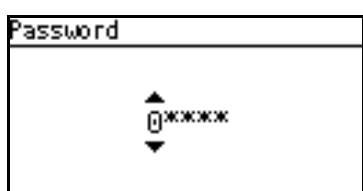


il PIN è parzialmente mascherato per evitare che venga visto da estranei, ma mentre tutte le altre cifre vengono visualizzate come asterischi, quella attualmente in corso di inserimento viene lasciata in chiaro: potete quindi usare i tasti ▲▼ per modificare ciascuna singola cifra (solo su X2, potete 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, ZP1/ZP2 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. [113](#)), 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. [83](#)).

10.5 MENU SUPERVISORE

Premete **.F + ▲** per visualizzare il prompt di richiesta password operatore:



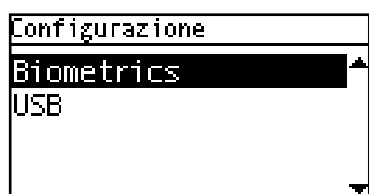
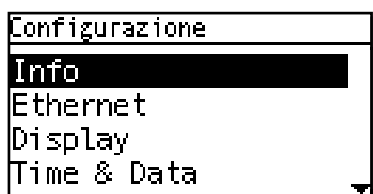
La password è parzialmente mascherata per evitare che venga vista da estranei, ma mentre tutte le altre cifre vengono visualizzate come asterischi, quella attualmente in corso di inserimento viene lasciata in chiaro. Il primo valore mostrato per ciascuna cifra è sempre '0', per cui essendo "00000" la 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. [106](#).

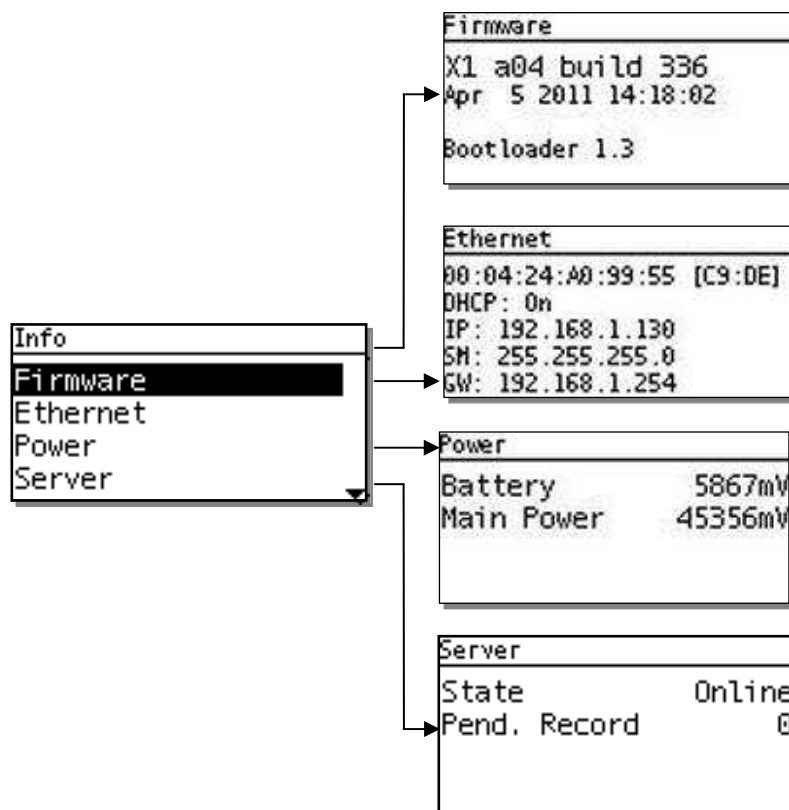
Nota: dopo 30 secondi di inattività, ZP1/ZP2 esce automaticamente dal menu supervisore.

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 4 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):

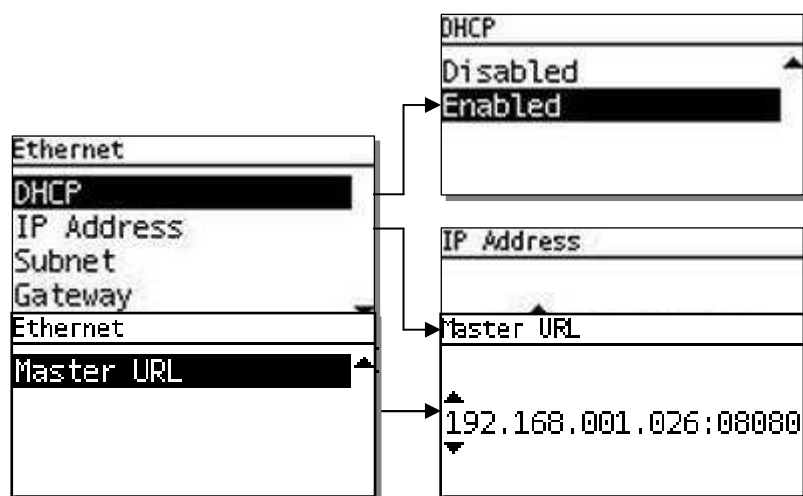


Il sottomenu “Info/Firmware” mostra l’attuale versione e la data di rilascio del firmware, e anche la versione di Bootloader.

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.

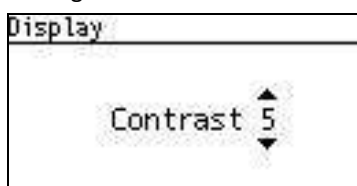
Il menu “Ethernet” contiene 5 sezioni, una per ciascun parametro impostabile: modalità DHCP, indirizzo IP (usato solo se il DHCP è disabilitato, o nel caso in cui il server DHCP non risponda), subnet mask, indirizzo gateway e (visibile solo spostando la selezione verso il basso fino alla pagina successiva) Master URL:



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.

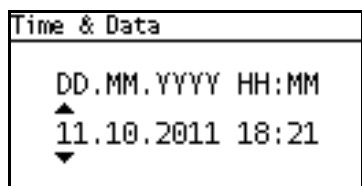
Nota: ciascun byte dell'indirizzo IP va sempre specificato in decimale su 3 cifre con eventuali zeri di riempimento a sinistra; i punti di separazione fra i byte sono fissi e vengono automaticamente saltati quando si procede all'impostazione della cifra successiva. Nel solo parametro MasterURL, il carattere ":" 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.

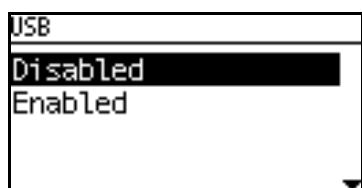
La voce "Time & Date" consente di impostare data e ora manualmente, utile nel caso in cui ZP1/ZP2 venga utilizzato come terminale *standalone* senza un collegamento Ethernet (il che rende impossibile l'impostazione dell'ora tramite il web server HTTP o tramite il caricamento di un file via FTP, come 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. 98.

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. 120): in pratica permette di impostare il parametro **Enable** all'interno della sezione [USB] del file PARAMETERS.TXT (vedi §4.10 a pag. 45) 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. 120), il quale comparirà solo in seguito all'inserimento della chiavetta una volta abilitata la gestione.

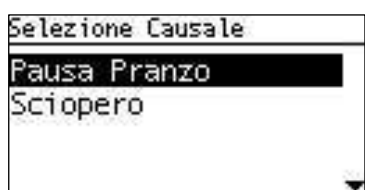


Usate i tasti freccia ▲▼ per selezionare un'opzione e ↵ (Enter) per confermare.

In ogni schermata potete premere il tasto [←]→ per abortire e tornare immediatamente alla schermata precedente, e infine uscire dal menu supervisore.

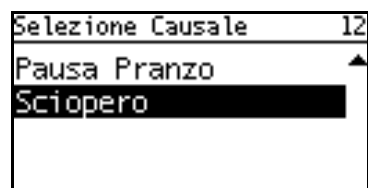
10.6 TRANSAZIONI CON CODICE CAUSALE

Se è stato precedentemente caricato un file di testo chiamato REASONS.TXT (vedi §4.4 alla pag. 17), o in alternativa un altro file chiamato AXREASON.TXT (vedi §5.10 alla pag. 63), 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):



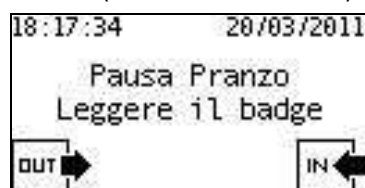
Nota: tenendo premuti i tasti ▲▼ è possibile scorrere velocemente la lista in entrambi i sensi (modalità *auto scroll*).

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. 18), 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. 83. 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:

```
18:18:45    20/03/2011
Pausa Pranzo
Uscita: 000646
```

Nel caso in cui si usi il file AXREASON.TXT, il controllo degli accessi sia attivato (vedi §5 a pag. 52), e siano stati caricati i file CARDS.TXT (§5.4 a pag. 55) e USERS.TXT (§5.9 a pag. 61), 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:

```
15:19:32    30/09/2011
Pausa Pranzo
Mario Rossi
Uscita
```

10.7 REVISIONE DATI DI PRESENZA

Dallo schermata di stand-by premete il tasto “freccia su” (▲):

```
Revisione Dati Pres.
Leggere il badge
```

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 §4.10 a pag. 24), 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 ▲▼:

```
Cod: 000043
30/09 11:28 Ent
30/09 11:41 Usc Pausa Pr
30/09 11:47 Ent
30/09 11:48 Usc
30/09 11:49 Ent
```

Nota 1: tenendo premuti i tasti ▲▼ è possibile 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. 75).

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. 17) o, in alternativa, nel file AXREASON.TXT (vedi §5.10 alla pag. 63):

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:


```

Cod: 000043
30/09 11:28 Ent
30/09 11:41 Usc      602
30/09 11:47 Ent
30/09 11:48 Usc
30/09 11:49 Ent
  
```

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 §4.10 a pag. 24) 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 §4.10 a pag. 24) 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.

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. 18), 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 ZP1/ZP2 viene gestito dal programma XatI@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. 91: le voci si possono scorrere con i tasti ▲▼ e selezionare col tasto ↵ (Enter) (c’è un timeout di 10 secondi).

```

Selezione Causale
Pausa Pranzo
Sciopero
  
```

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

ZP1/ZP2 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 §4.10 a pag. 37. La pagina **"Biometrics"** del web server HTTP ha l'aspetto mostrato in figura:

X1/X2 Configuration

- Network
- GPRS modem
- FTP Client
- Time & Attendance
- Access Control
- Reader 1
- Reader 2
- External Reader
- Daylight Saving Time
- Time and Date
- USB
- System
- Remote Relays
- Biometrics
- File Manager
- Password
- Log

Biometrics

Enabled ☒

Firmware V1.9F (SFM3000-SFM4000 series)

Template number 2

Available Finger 9588

Enrolled users 0

Sensor type STMicromicro TouchChip

Auto scan mode ☐

Visitors free pass ☐

Enroll only authorized ☐

Enroll all ☐

Security level Automatic Normal

Sensor sensitivity 8 - Most Sensitivity

Image quality Moderate qualification

Lighting condition ☐ Indoor ☒ Outdoor

Fast Mode 7 - Automatic

Mifare first block 1

Minimum fingerprint quality 70 %

Template source On card or on terminal

Export archive

Delete all templates

Calibrate sensor

Sensor parameters dump

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, ch   il limite massimo, meno il numero di *template* gi  memorizzati)
- il numero degli utenti biometrici gi  registrati (per ciascun utente   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 modalit  "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   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 gi  presenti nel modulo. Per questo motivo si pu  anche parlare di modalit  "solo dito".

Nella modalit  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   impostato a 1, vedi §4.10 a pag. 24) 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 identit  confrontando il *template* appena scansionato solo con quelli relativi al codice tessera introdotto.

Nota: in modalit  "autoscan"   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   prevista la registrazione nel sistema biometrico) dalla richiesta di verifica 1:1 (corrispondente al parametro **FreePass**, vedi ulteriori dettagli al §11.4 a pag. 110)
- 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. 55)
- la *checkbox* per consentire la registrazione di impronte per tutti i codici tessera inseriti (corrispondente al parametro **EnrollAll**). Normalmente invece, se   presente almeno un file fra CARDS.TXT e CARDNRNGE.TXT, il codice inserito viene accettato solo se   fra quelli elencati in CARDS.TXT o si trova all'interno di un intervallo di codici elencato in CARDNRNGE.TXT.
- il menu a tendina per impostare il livello di sicurezza del modulo biometrico, corrispondente al parametro **SecurityLevel**. Tale parametro pu  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: pi  sicuro)

Il livello di sicurezza specifica il FAR (*False Acceptance Ratio*, cio  rapporto di falsa accettazione): valori di sicurezza pi  alti corrispondono a valori FAR pi  bassi. Ad esempio un FAR di 1/100.000 significa che la probabilit  di accettare impronte non autorizzate   pari a 1/100.000. Poich  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	1/1,000,000,000

- 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.
- i *radiobutton* per impostare le condizioni di luminosità ambientale (utilizzo all’aperto o al chiuso), corrispondenti al parametro **LightingCondition**. Tale parametro può assumere solo i valori 0 (utilizzo all’aperto - default) o 1 (utilizzo al chiuso), e può avere effetto solo nel caso di sensore di impronte di tipo ottico, in quanto la luminosità dell’ambiente circostante (incidenza dei raggi luminosi, passaggi da luce a ombra) può influire sulle prestazioni del sensore, generando ad esempio delle scansioni “fantasma” in assenza del dito, o non rilevandone la presenza.
- 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. [107](#)): 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. [107](#)), 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. [105](#))
- "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. [106](#))

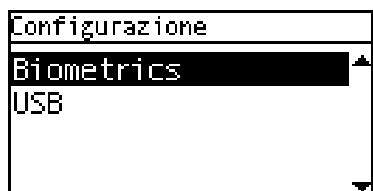
- “Calibrate sensor”: ricalibrazione del sensore, da effettuare nel caso si verificano peggioramenti nelle prestazioni del sensore (ad esempio diversi utenti che in precedenza venivano riconosciuti senza problemi ad un certo punto non lo sono più, vedi anche §11.1 a pag. [106](#))

Infine è disponibile un pulsante utilizzabile a solo scopo di debug:

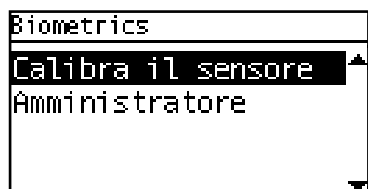
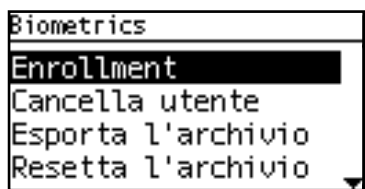
- “Sensor parameters dump”: scrittura del valore di tutti i parametri del modulo biometrico all'interno del file LOG.TXT

11.1 MENU DI GESTIONE DELL'ARCHIVIO DELLE IMPRONTE

L'archivio delle impronte contenuto nel modulo biometrico può essere gestito direttamente dalla console di ZP1/ZP2 mediante la sezione “Biometrics” del menu supervisore descritto al §10.5 a pag. [88](#).



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 §4.10 a pag. [33](#)) sia impostato ai valori '30' o '32' (quelli relativi ad un lettore RFID2 seriale TTL 13,56MHz), la sezione “Biometrics” contiene una ulteriore voce “Canc tessera Mifare”: vedi §11.2 a pag. [107](#) 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. [106](#)).

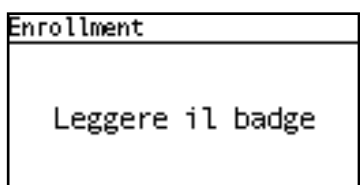
Nota 3: solo in caso di sensore di impronte di tipo ottico, la voce “Calibrate” non compare fra le opzioni disponibili in quanto non necessaria.

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 ZP1/ZP2 chiede di leggere il badge dell'utente che deve registrare le sue impronte (timeout: 10 secondi):

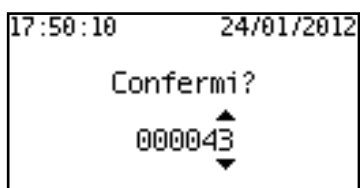


In alternativa, è sempre possibile inserire manualmente il codice utente:

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).

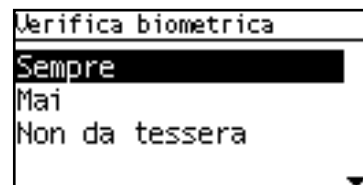
In tutti i casi sopra citati è anche possibile abortire direttamente mediante i tasti **.F** o **[<-]->**.

Una volta confermato il codice, ZP1/ZP2 controlla se sia stato caricato almeno un file fra CARDS.TXT e CARDRNGE.TXT (vedi §5.1 a pag. 52), e se li trova verifica che il codice inserito sia fra quelli elencati in CARDS.TXT 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 §4.10 a pag. 39) 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 §4.10 a pag. 39): in questo caso verranno sempre accettati tutti i codici inseriti.

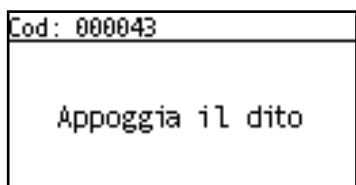
Se il codice viene accettato e non risulta essere già presente nessuna impronta associata al codice inserito, appare la schermata per la selezione della modalità di richiesta della verifica biometrica 1:1 per quel codice (timeout: 30 secondi; vedi anche al §11.4 a pag. 110 le ulteriori modalità di esenzione 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 almeno uno dei parametri **CardDecode** all'interno delle sezioni *[Reader1]*, *[Reader2]* e *[ExtReader]* del file PARAMETERS.TXT (vedi §4.10 a pag. 33) sia impostato ai valori '30' o '32' (quelli relativi ad un lettore RFID2 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. 107 per ulteriori dettagli.

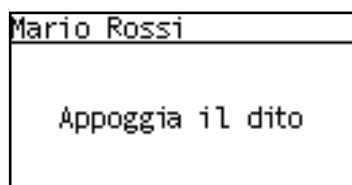
Selezionando "Sempre", ogni volta che nello stato di attesa transazioni si leggerà questa tessera o si digiterà manualmente questo codice (solo su X2 e se il parametro **AllowTypeCode**=1 all'interno della sezione *[TimeAttendance]* del file PARAMETERS.TXT, vedi §4.10 a pag. 24) verrà richiesto di appoggiare il dito sul sensore per procedere alla verifica biometrica (con le eccezioni descritte al §11.4 a pag. 110), 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. 95.



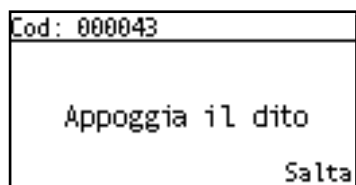
Il passo successivo è quello relativo alla scansione dell'impronta: ZP1/ZP2 mostra in alto a sinistra il codice inserito, oppure il nome dell'utente nel caso sia possibile risalirvi (cioè solo se sono stati caricati entrambi i file CARDS.TXT e USERS.TXT, a prescindere dal fatto che il controllo accessi sia attivato o meno), e chiede di appoggiare il dito sul sensore di impronte digitali (timeout: 10 secondi):



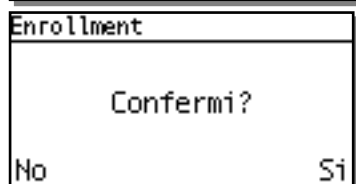
oppure



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) (Sì) 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:

Cod: 000043
Appoggia di nuovo

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):

Enrollment
Confermi?
Qualità template: 100%
No Si

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

\$4.10 a pag. 39 (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:

Enrollment
Inserire nuovo dito?
No Si

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. [105](#)), 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. [104](#)):

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. [106](#)). 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. [105](#) che modificando direttamente il file **USERCODS.TXT** (attenzione: in questo caso leggete la nota a pag. [104](#)):

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^(*): se il controllo accessi è abilitato, questo campo può essere utilizzato per

validare le transazioni in modalità “solo dito” (identificazione 1:N), confrontandolo con l’analogo campo **R** all’interno dei record del file CARDS.TXT (vedi §5.4 a pag. 55 per ulteriori dettagli).

- 0: provenienza della lettura indifferente (solo per registrazioni di impronte effettuate su versioni di firmware precedenti alla a07_build863: valore gestito per retrocompatibilità)
- 1: tessera letta sul lettore primario (READER1) o codice inserito manualmente
- 3: tessera letta sul lettore esterno su morsetteria 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 USERCODES.TXT.

In seguito ad ogni registrazione di impronta, oltre ad aggiornare il file USERCODES.TXT, ZP1/ZP2 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 USERCODES.TXT. Lo scopo di BIOUPDATE.TXT è consentire la sincronizzazione degli archivi di impronte contenuti nei moduli biometrici FingerBOX di altri terminali ZP1/ZP2 “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. 122) di questo file dal terminale “master” agli “slave” (e conseguente importazione automatica dei dati, vedi §11.3 a pag. 109), 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 USERCODES.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.

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<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. 109)

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)

<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 di impronte: cancellazione di un utente, cambiamento degli attributi di un utente (esenzione dalla verifica biometrica, 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 <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 cambiamento degli attributi dell'utente.

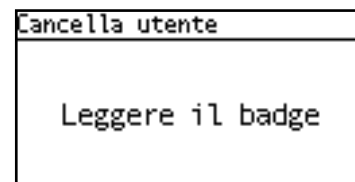
Tutti gli altri campi rimangono immutati.

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. ZP1/ZP2 chiede solo di leggere il badge dell'utente che deve essere cancellato (timeout: 10 secondi):

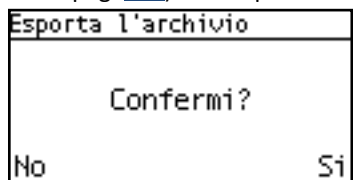
In alternativa, è anche possibile inserire manualmente il codice utente, in una delle 2 modalità già descritte nel caso della registrazione delle impronte (vedi pag. 99).



In questo caso non viene chiesta nessuna conferma: ZP1/ZP2 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. 94, o tramite caricamento di un file apposito via FTP, vedi §16 a pag. 128). Come prima cosa ZP1/ZP2 chiede conferma dell'operazione (timeout: 30 secondi):



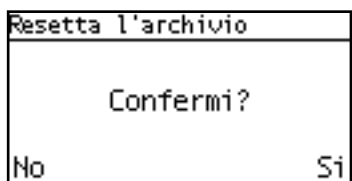
Premete ↵ (Enter) (Si) per procedere e [←]-> (No) per abortire. L'esportazione dell'archivio consiste nella creazione di un file di testo **BIODATA.TXT** all'interno della cartella \BIOEXP (normalmente non presente ma creata, assieme alla cartella \BIOIMP, subito dopo avere abilitato la gestione del 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 ZP1/ZP2 "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. 122) di questo file dal terminale "master" agli "slave" (e conseguente importazione automatica dei dati, vedi §11.3 a pag. 109), 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. 94, o tramite caricamento di un file apposito via FTP, vedi §16 a pag. 128). Come prima cosa ZP1/ZP2 chiede conferma dell'operazione (timeout: 30 secondi):



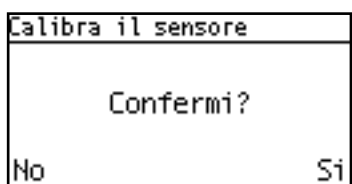
```

Resetta l'archivio
Confermi?
No                Si
  
```

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 **USERCODES.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. 94). Come prima cosa ZP1/ZP2 chiede conferma dell'operazione (timeout: 30 secondi):



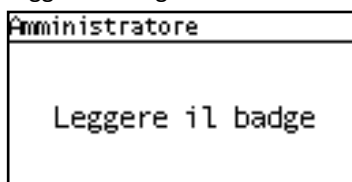
```

Calibra il sensore
Confermi?
No                Si
  
```

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. 94), 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 ZP1/ZP2 chiede di leggere il badge dell'utente che si vuole definire come amministratore (timeout: 10 secondi):

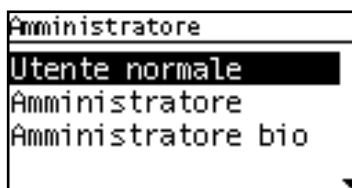


```

Amministratore
Leggere il badge
  
```

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. 99).

Qualunque sia il codice tessera inserito, ZP1/ZP2 mostra a questo punto la schermata di selezione degli attributi dell'utente (timeout: 30 secondi):



```

Amministratore
Utente normale
Amministratore
Amministratore bio
  
```

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, ZP1/ZP2 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):

```
Amministratore
Confermi?
No      Si
```

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):

```
Password
  ▲
  ▼
  0****
```

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. 94), 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. 88, 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. Solo gli 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.

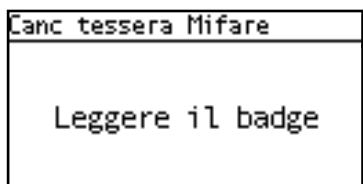
11.2 SALVATAGGIO DELLE IMPRONTE SU CARTE MIFARE

Come visto al §11.1 a pag. 98, nel caso in cui almeno uno dei parametri **CardDecode** all'interno delle sezioni [Reader1], [Reader2] e [ExtReader] del file PARAMETERS.TXT (vedi §4.10 a pag. 33) sia impostato ai valori '30' o '32' (quelli relativi ad un lettore RFID2 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. 98).

```
Biometrics
Enrollment
Cancella utente
Esporta l'archivio
Resetta l'archivio
```

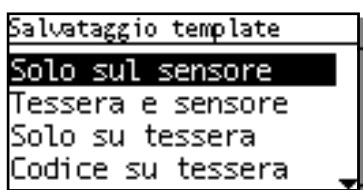
```
Biometric
Canc tessera Mifare
Calibra il sensore
Amministratore
```

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, ZP1/ZP2 chiede solo di posizionare sul lettore la carta Mifare che deve essere cancellata (timeout: 10 secondi):



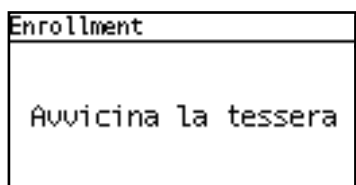
Appena rilevata la carta Mifare, ZP1/ZP2 passa alla sovrascrittura di tutti i blocchi dati coinvolti: questo processo richiede alcuni istanti (il terminale mostra il messaggio **"Attendere prego ..."** e la percentuale di avanzamento), quindi non si deve allontanare la carta dal lettore fino alla visualizzazione 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. 100, 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 §4.10 a pag. 38) 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. 100, 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 §4.10 a pag. 38): 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. 33).

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 USERCODES.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 “Sì” 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. 33).

11.3 IMPORTAZIONE DI IMPRONTE NEL MODULO BIOMETRICO VIA FTP

Come abbiamo visto al §11.1 a pag. 98, 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 ZP1/ZP2 “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. 122) 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 ZP1/ZP2 “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). ZP1/ZP2 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.

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 succedrebbe 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, contenente 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 \BIOIMP 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 ZP1/ZP2). 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 ZP1/ZP2, 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 ZP1/ZP2, 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.

ZP1/ZP2 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. [103](#)).

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 §4.10 a pag. 39, o analogamente spuntare la checkbox **"Visitors free pass"** nella pagina **"Biometrics"** del web server HTTP del terminale, vedi §11 a pag. 94. 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 §4.10 a pag. 24) 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 §4.10 a pag. 36; 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).

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 §4.10 a pag. 42) oppure, analogamente, selezionando il radio button **"HTTP"** fra le opzioni **"Protocol"** nella pagina **"Network"** del web server HTTP.

In questo caso, impostando la modalità online (si veda il parametro **Offline** nella sezione *[TimeAttendance]* del file PARAMETERS.TXT, §4.10 a pag. 25), il terminale ZP1/ZP2 invia un messaggio HTTP GET al **MasterUrl** (vedi PARAMETERS.TXT) ad ogni transazione, 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**.

12.1 MESSAGGI HTTP PER TRANSAZIONI ONLINE (DA ZP1/ZP2 A MasterURL)

Il messaggio di transazione online viene costruito concatenando i seguenti 2 parametri: **MasterUrl** e **httpOnlineMessage**.

MasterUrl contiene l'indirizzo web del server, o il suo indirizzo IP
(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 memorizzato nel file TRANSACTIONS.TXT
- **\$fullcode\$**
se presente, viene rimpiazzato con l'intero codice restituito dal lettore di badge, 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**
- **\$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)
- **\$batt\$**
se presente, viene rimpiazzato con l'attuale stato di carica della batteria (vedi §13.1 a pag. [119](#))
- **\$battmV\$**
se presente, viene rimpiazzato con l'attuale valore in mV della tensione sulla batteria

Esempio:

MasterUrl=http://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. [114](#)), che vengono inviati al server periodicamente per segnalare la presenza del file TRANSACTIONS.TXT.

12.2 FORMATO RISPOSTA DEL SERVER (DA MasterURL A ZP1/ZP2)

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 §4.10 a pag. [24](#))

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. [52](#)). 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 1: 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. [118](#)), o ad un messaggio "Keep Alive" (vedi §12.3 qui sotto).

Nota 2: questa funzionalità è ancora in fase sperimentale, pertanto sul display, contestualmente alla richiesta di introduzione del PIN, compaiono anche il PIN atteso e, se specificato, l'eventuale PIN di accesso sotto minaccia.

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 ZP1/ZP2 A MasterURL)

ZP1 e ZP2 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. 111, 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), ZP1/ZP2 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 ZP1/ZP2 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 ZP1/ZP2)

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 il campo “save” non ha senso e quindi viene ignorato; 2) il campo “pin=” non ha senso e quindi non deve essere usato; 3) in questo caso è invece possibile usare due ulteriori campi che non vengono gestiti nella risposta ad una transazione online:

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”, ZP1/ZP2 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.

CONSIDE [<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 §4.10 a pag. 25).

RECOVER gg mm aaaa

Forza la ritrasmissione immediata, in modalità "batch" (vedi §12.5 a pag. 118), 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. 92) 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. 75).

Quando si invia questo comando in risposta ad un "Keep Alive", ZP1/ZP2 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 §4.10 a pag. 23). 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. 94). 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", ZP1/ZP2 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.

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 §4.10 a pag. 23), 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", ZP1/ZP2 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".

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. 102): se quest'ultimo dovesse valere '0' (solo per registrazioni di impronte effettuate su versioni di firmware precedenti alla a07_build863), 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. 94). **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**.

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" ZP1/ZP2 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).

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=CONSIDLE "Attendere 10 secondi" 10

cmd=RECOVER 1 12 2011

cmd=GETPAR [TimeAttendance] DirMode

cmd=SETPAR [TimeAttendance]TAB TABDirMode=3TAB TAB[Reader1]TAB TABCardDecode=30TAB

TABCardCodeLength=10

cmd=BIODEL 1_0000000000123456

cmd=BIORESET

file=PROVA,14

12345

67890

12.5 MODALITA' ONLINE: SERVER NON IN LINEA

In modalità online ZP1/ZP2 invia un messaggio HTTP GET al MasterURL ad ogni transazione, come spiegato nel §12.1 a pag. 111.

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 non ancora marcata come "già ricevuta dal server" al MasterURL, 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 i record precedentemente registrati 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=http://www.yourserver.com:8181
```

Appena ricevuta la risposta al "Keep Alive", ZP1/ZP2 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 ZP1/ZP2 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.

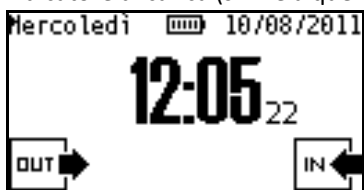
Nota 2: Rispondere ad una transazione offline ricevuta in modalità "batch" con una HTTP RESPONSE contenente la richiesta di introduzione PIN (*pin=*) non ha ovviamente senso.

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).

13. FUNZIONAMENTO A BATTERIA

ZP1 e ZP2 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, §4.10 a pag. 39, 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 §4.10 a pag. 39) al valore 0 (il default è 1), o deselectando 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. 127), 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 §3.8 a pag. 12) ZP1 e ZP2 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.

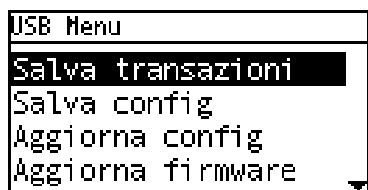
14. UTILIZZO DELLA CHIAVETTA USB

A partire dalla versione di hardware **006** (vedi §3.8 a pag. [12](#)) ZP1/ZP2 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 §4.10 a pag. [45](#), 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. [88](#)).

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.

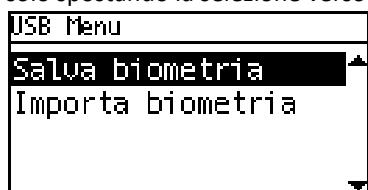
Una volta attivata questa funzionalità, inserendo una chiavetta di memoria USB nell'apposito connettore dopo avere rimosso il cappuccio in gomma (vedi figura 4 al §3.1, pag. [7](#)), comparirà una schermata di richiesta password identica a quella per l'accesso al menu supervisore (vedi §10.5 a pag. [88](#)), 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 ZP1/ZP2 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:



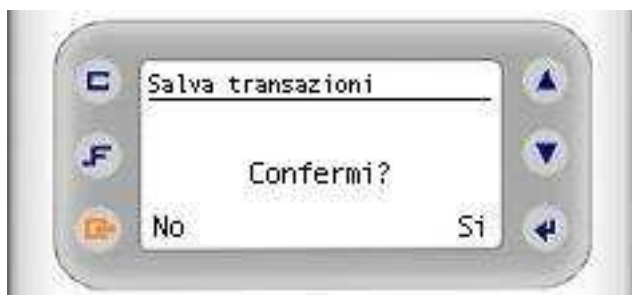
Si veda ai §14.5 e §14.6 a pag. [122](#) quali sono le condizioni affinché queste voci siano disponibili.

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 §4.10 a pag. [25](#) 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 deselectare 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 utilizzare in 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.

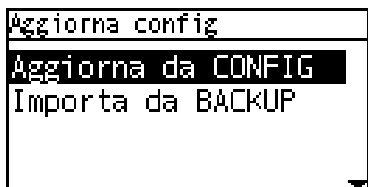
14.3 CARICAMENTO CONFIGURAZIONE DA CHIAVETTA USB

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:

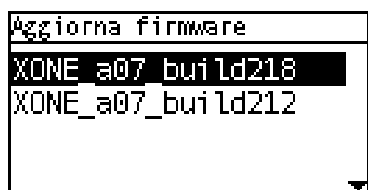


Usate i tasti freccia **▲▼** per selezionare una voce di menu e **↵** (Enter) per confermare.

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_ann_buildxxx.bin**” contenente la nuova versione di firmware (vedi §9 a pag. 81). 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:

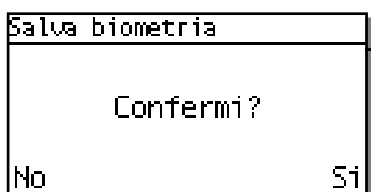


Usate i tasti freccia ▲▼ per selezionare quale versione di firmware caricare ↵ (Enter) per confermare. Anche per questa operazione compare l’ulteriore finestra di richiesta conferma già descritta in precedenza, ma al termine dell’operazione il terminale si riavvia automaticamente col nuovo firmware, uscendo quindi dal menu USB.

14.5 SALVATAGGIO DATI BIOMETRICI SU CHIAVETTA USB

Se ZP1/ZP2 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. 98), 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) (Sì) 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 ZP1/ZP2 “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. 109), 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. 109. 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 ZP1/ZP2 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 ZP1/ZP2): 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 **USERCODES** associato (se anch'esso presente nella cartella **\BIOEXP**).

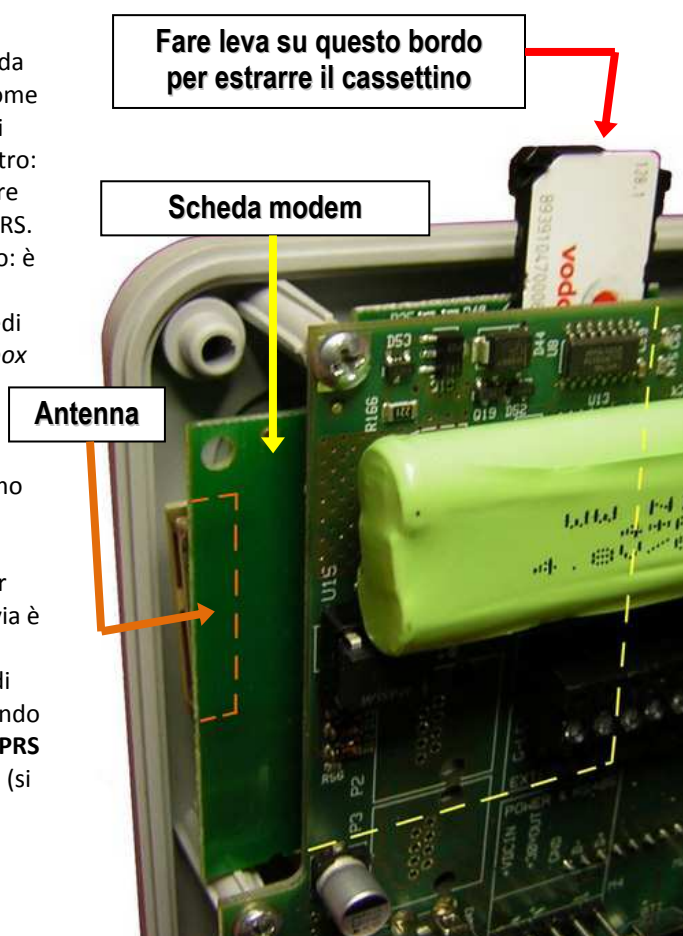
Tutte le opzioni chiedono conferma prima di procedere: premete ↵ (Enter) (Si) per procedere e [←]-> (No) per abortire.

Importa biometria	
Confermi?	
No	Si

15. USO DEL MODEM GPRS OPZIONALE

ZP1/ZP2 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 cassetto 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 §4.10 a pag. 43) 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 §4.10 a pag. 43):



Oltre alla *checkbox* già spuntata per l'attivazione della gestione del modem GPRS (corrispondente al parametro **Enabled**) compaiono, nell'ordine:

X1/X2 Configuration

- Network
- GPRS modem
- FTP Client
- Time & Attendance
- Access Control
- Reader 1
- Reader 2
- External Reader
- Daylight Saving Time
- Time and Date
- USB
- System
- Remote Relays
- Biometrics
- File Manager
- Password
- Log

GPRS modem

Enabled ☒

IMEI code 357460-03-266068-5

Connection status Disconnected

Signal quality -77 dBm

Connection interval 0 minutes (set 0 to stay always connected)

AT extra command

Dial number

User

Password

Primary DNS 0 . 0 . 0 . 0

Secondary DNS 0 . 0 . 0 . 0

Reset GPRS module

Send AT command

- il codice IMEI (*International Mobile Equipment Identity*) che identifica univocamente il modem GPRS
- 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**): il valore di default (0) significa che 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). Se impostato ad un valore diverso da 0, invece, 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 (ZP1/ZP2 gestito da XAM o da XatI@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. 113), 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.

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. 79), 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”: in questo caso il modem effettuerà la connessione GPRS solo se vi sono esportazioni schedulate, e solo in corrispondenza degli orari impostati, disconnettendosi automaticamente al termine di ciascuna sessione FTP.

- 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 seguente:

991#**

- 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 §4.10 a pag. 42 e 44). 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 ZP1/ZP2, 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)

(*) **Nota importante:** se viene collegato un cavo Ethernet al connettore RJ45 di ZP1/ZP2, la connessione GPRS non viene mai effettuata. Nel caso in cui il cavo Ethernet venga collegato a connessione GPRS in corso, quest'ultima verrà subito interrotta.








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 ZP1/ZP2 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. 119): 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		Tentativo di connessione GPRS in corso
	Segnale con potenza minore o uguale a -95dBm		Connessione GPRS stabilita
	Segnale con potenza compresa fra -93dBm e -73dBm		Mancanza parametri di connessione oppure modem bloccato (stato temporaneo)
	Segnale con potenza maggiore o uguale a -71dBm		

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 ZP1/ZP2 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. [105](#)). 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. [94](#).

- **\$BIORESET.CMD**

Se ZP1/ZP2 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. [106](#)). 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. [94](#).

- **\$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. [75](#).

- **\$RECOVERggmmaaaa.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. 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. [114](#) per approfondire il concetto di messaggio **"Keep Alive"**, ed il §12.4 a pag. [114](#) 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 ZP1/ZP2 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://partnersarea.axessworld.it/tmcarearea/area/area.asp>) potete scaricare "X1HTTPEdemo": 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 ZP1/ZP2 in modalità online (**Attenzione:** riconosce solo le transazioni in formato standard, vedi §7 a pag. 75).

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 ZP1/ZP2). 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>

18. MAPPE DEI CARATTERI

ZP1/ZP2 può visualizzare tutti e soli i caratteri inclusi nelle tabelle di codifica Windows-125x relative ai seguenti set di caratteri: Europa occidentale (Windows-1252), Europa centro-orientale (Windows-1250), Turco (Windows-1254). Il set di caratteri (e quindi la tabella di codifica utilizzata) si può selezionare impostando il valore del parametro **FontEncoding** all'interno della sezione [System] del file PARAMETERS.TXT (vedi §4.10 a pag. 40), il cui valore di default è 0, che corrisponde al set di caratteri Europa occidentale. Vedete nella pagina seguente quali sono i valori del parametro **FontEncoding** relativi agli altri set di caratteri disponibili. Il set di caratteri può anche essere impostato dalla pagina "System" del web server HTTP del terminale, dove è presente un menu a tendina contenente la descrizione di tutti i set di caratteri disponibili:

X1/X2 Configuration

Network	System
GPRS modem	Firmware X1 a08 build 101, May 23 2012 17:32:15
FTP Client	Bootloader 1.4
Time & Attendance	MAC Address 00:04:24:A1:DC:2B [DD,A3]
Access Control	Available Free Space 1894 MBytes
Reader 1	Battery 5764 mV - Normal
Reader 2	Server Offline - Pending Record 6
External Reader	Screen Snapshot <input type="button" value="Snapshot"/>
Daylight Saving Time	Restart Terminal <input type="button" value="Restart"/>
Time and Date	Format SD Card <input type="button" value="Format"/>
USB	Reset default parameters <input type="button" value="Reset"/>
System	Recover all the transactions <input type="button" value="Recover"/>
Remote Relays	Language <input type="button" value="English"/>
Biometrics	Font encoding <input type="button" value="Western European - Windows-1252"/>
File Manager	Audio volume <input type="button" value="Western European - Windows-1252"/>
Password	Timeout on Battery <input type="button" value="Central European - Windows-1250"/>
Log	Turn Off Backlight on Battery <input checked="" type="checkbox"/>

Nella pagina seguente sono mostrate le tabelle di codifica Windows-125x relative a ciascun set di caratteri (in cui sono riportati gli indici dei caratteri in valore esadecimale); i punti blu 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	€	.	,	f	„	...	†	‡	^	%	Š	<	£	.	Ž	.
9	.	‘	’	“	”	•	—	~	™	š	>	œ	.	.	Ÿ	
A	;	ç	£	¤	¥	!	\$	~	©	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)
FontEncoding=0 (default)

	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	€	.	,	f	„	...	†	‡	^	%	Š	<	£	.	.	
9	.	‘	’	“	”	•	—	~	™	š	>	œ	.	.	Ÿ	
A	;	ç	£	¤	¥	!	\$	~	©	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	Ġ	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	ß	
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ġ	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	ı	ÿ

Turco (Windows-1254)
FontEncoding=1

	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	<	Š	Ť	Ž	Ž	
9	.	‘	’	“	”	•	—	.	™	š	>	š	ť	ž	ž	
A	˘	˘	˘	˘	˘	˘	˘	˘	˘	˘	˘	˘	˘	˘	˘	˘
B	°	±	²	³	´	µ	¶	·	.	¹	º	»	¼	½	¾	¿
C	Á	Â	Ã	Ä	Å	Ł	Ć	Č	È	É	Ê	Ë	Ì	Í	Î	Ď
D	Đ	Ň	Ů	Ó	Ô	Õ	Ö	×	Ř	Ú	Û	Ü	Ý	Ť	ß	
E	á	â	ã	ä	å	ł	ć	č	è	é	ê	ë	ì	í	î	ď
F	đ	ň	ů	ó	ô	õ	ö	÷	ř	ú	û	ü	ý	ť	ı	ÿ

Europa centro-orientale(Windows-1254)
FontEncoding=2